1.95 on diskette

Presents . . .

TRAKCESS

Written by: **Roxton Baker**

TRAKCESS

"Access to Tracks"

Written by:

Roxton Baker
P.O. Box 8308
APO San Francisco, 96555

Congratulations! You have just purchased THE MOST POWERFUL TRS-80 disk access utility yet written. Perhaps you were able to share the cost with a few friends, but even if not I hope you will find Trakcess worth the price. Should you discover any problems with the program, or have suggestions for improvement, please contact either TAS or write to me. User feedback is one way that good programs get better. Your input will be appreciated.

INTRODUCTION

Several fine disk access programs are currently available, such as Superzap, Z80ZAP, and some newer ones. For zapping standard TRSDOS disks they are excellent, featuring quick access to a file's sectors and convenient 'paging' through them. Trakcess is not intended to replace these utilities. It is unable to interpret a disk's directory, and since it makes no distinction between "standard" sectors and any others, it cannot offer the pre-formatted display and easy file I/O of, say, Superzap. Trakcess was written with different goals:

> To bring under direct user control, with as much software assistance as is practical, every capability of the TRS-80's 1771 Floppy Disk Controller (FDC).
>
> To combine these capabilities into powerful "intelligent" functions whose only limitations are those of the machine in general, or of the FDC in particular.

The inspiration for this work was Bill Barden's excellent and unique "Disk Interfacing Guide for the TRS-80" which will hopefully, upon the advent of Trakcess, find renewed interest among serious disk users. The Alternate Source will have copies of this book available. Certainly you will need it if you are not already an expert on the FDC. This documentation will briefly discuss the elements of track formatting and sector identification, but AT A MINIMUM the non-expert will require a copy of the Western Digital data sheet on the 1771 FDC. This can be found in the Expansion Interface Service Manual (Radio Shack catalog number 26-1140). Or you may persuade WD to send you one. National Semiconductor, which is a second source for the 1771, also has a useful data sheet on it.

Trakcess is powerful precisely because it works on the most elementary levels. It has been padded around with lots of interactive software and utility functions so that it remains fairly efficient to use, but it is not the tool of choice for simple zapping. Trakcess assumes very little about the disks it is used on. As a result, there are often many questions for the operator to answer. To make full use of Trakcess you must be prepared to study the literature and experiment with the possibilities. I recommend taking a few track dumps of disks you have on hand and comparing what you find there to what the books say. Then try making up your own tracks and formatting a blank diskette with them. Trakcess puts all of this at your fingertips. Beyond this, you might exercise the disk identification features such as (S)can Track Sectors and (L)ocate Disk Sectors. These commands embody some routines that are simple conceptually, but that were quite difficult to write. They are far more potent than they seem and form the heart of the highest level commands in Trakcess, such as (C)opy Track and (D)uplicate Disk. These latter are capable of analyzing many widely different disk formats and duplicating them, and are far removed from the primitive FDC functions. However, in the event that they require your assistance, you will still have to know what you're doing.

USING TRAKCESS

Trakcess requires a 48K machine. The (C) and (D) commands require two drives. The program has been tested under NEWDOS and TRSDOS, but should work under any DOS. Trakcess consists of:

TRAKCESS (a Basic program), and
TRAKCESS/CMD (a machine language file).

To get started from DOS, just type 'TRAKCESS'. Then follow the prompts. At the query as to whether a printer is attached and ready, your Y or N response will determine whether or not line-printer output options will be offered to you later. After that, the menu will come up. IMPORTANT NOTE: at almost any time when the program is awaiting input or is executing a lengthy command, pressing 'M' will return you to this menu.

The following is a description of each of the menu commands available. Please read this at least briefly before running the program. Trakcess, like any other disk utility, can easily destroy valuable data on a disk if you aren't reasonably careful. ALWAYS write-protect your master disks, and work on back-up copies!

Some terminology used in these descriptions will not be familiar to you. Definitions and explanations will be found in the DISK STRUCTURE AND FORMATTING section.

1. Select Drive

You must tell Trakcess which of the four drives is to be the currently active one. You'll be asked its head position. Normally, you won't know, so just press ENTER. If the drive was previously selected, Trakcess will recall its last position, otherwise the head will be moved to track 0. If you enter a number here, Trakcess will assume that is the actual head position of that drive. NOTE: no other disk-related commands will work until a drive has been selected!

Trakcess contains all of the disk I/O routines that it requires, so once it is running you do not need a DOS disk in drive 0. However, if Trakcess stops on a program error, put a DOS disk in drive 0 before continuing in case Basic looks for one. Otherwise you may hang the system.

2. I and O

These step the head of the selected drive In and Out one track. "In" means closer to the center of the disk. These two keys repeat. Trakcess assumes thirty-five tracks. This is defined by DM (presently set at 34) near the beginning of the program. If you wish to work with a different number of tracks, change DM to that number minus 1.

3. (G)o to Head Position

Allows you to quickly position the head of the active drive at any track. Enter the desired track location in decimal.

4. (R)ead and (W)rite Sectors

These commands allow you to read into a specified block of memory, or write from it, any sector on the current track. You must specify the starting point in memory, the sector's ID bytes (TN and SN), its type (IBM or NBM) and its DAM (for a write). The length of the sector is of course already noted on the track, having been put there at format time. If a sector was previously specified Trakcess will allow you to reuse those specifications without reentering them. This facilitates modifying sectors and transferring them between disks.

One further feature of sector (R)eading is useful in the recovery of lost data. Trakcess will report an unsuccessful read and will give you the option of infinite retry. Unlike Superzap, Trakcess does not step the head all over the disk between attempts. That seems so hard on the drive. All sector (W)rites are verified by rereading, so data transfers under Trakcess are reliable.

5. (T)ake and (P)ut Tracks

These two commands involve the transfer of a whole track (about 3120 bytes) of information. A track read followed by a scan with Trakcess' editor will show you literally everything that is on the track. Do not be concerned if the number of bytes read in varies from time to time, this is normal for a track read.

The track write is usually used only for formatting a blank diskette with empty sectors. You can create any format track you want with the (B)uild command, then write it to disk with (P)ut. If none of the sectors on a track contain F7 bytes, it is feasible (though not always successful) to (T)ake a track from one disk and immediately (P)ut it onto another. Synchronization is a problem here, but it can work. In fact, sectors with F7 bytes can then be transferred separately using (R) and (W). Trakcess will do these things, but considers them poor form. You would only attempt to copy a track in this manner if you were desperate, and afterwards you wouldn't tell anyone.

6. (S)can Track Sectors or (L)ocate Disk Sectors

Trakcess will search the current track (or the whole disk) for all useable sectors. It will determine all the important information for each (TN, SN, SL, IBM/NBM, DAM) and will present it to the screen or printer. Do not be impatient; up to 30 seconds per track may be required for the scan. Try these commands on a standard TRSDOS disk, any "protected" disk, a CP/M disk, or even an Intecolor disk if you have one, and compare the results.

If a track has any "false" sectors (without a data record following, or with a bad ID pack) Trakcess will note the fact. If you are attempting to recover a damaged disk, you may be able to find some sector data using a track read, and transfer it to a track you have built using the (B)uild format track command discussed previously. Or the false sectors may have been purposely put on the disk as some sort of "protection". If you are attempting to duplicate such a "protected" diskette, use the (B) function and the editing technique outlined there.

Note that the (L)ocate command first tells you which tracks even have sectors, and then gives you the option of a full report.

7. (C)opy Track

You are prompted for a drive and track to copy to. You cannot copy to the active drive; another must be specified. Trakcess will search the current track and will identify all the sectors on it. Then it will build a matching format track in memory and write it to the target disk. Finally, it will transfer (with verify) all of the sectors. You are given the option of using the existing DAM's or specifying a new one. If any sectors on the source track are damaged or of indeterminable length, you will be prompted for the length you wish to make them. To decide on this, do a track read and look at that sector in memory. Try choosing the next larger multiple of 16 bytes than the sector's apparent length, unless you know what it should be.

8. (D)uplicate Disk

This command does nothing more than set up both drives and repeat the (C) command for each track. Most disks, "protected" or not, can be duplicated in this fashion, but it usually takes about 13 minutes. It's a good idea to check out the destination disk first by formatting it using TRSDOS, and then bulk-erasing it. Note that unlike Superzap, et. al., Trakcess can copy to a blank diskette as well as to a formatted one. If any sectors won't copy, note them and proceed. You may later be able to use (R) and (W) to transfer them, or that entire track may be retried with (C).

9. (E)dit or Fill Memory

This command allows you to edit memory with the scrolling, Electric Pencil-like editor utility that is built into Trakcess. Or you can fill memory between specified addresses with a specified byte. The editor requires a four-digit hex starting address. At the address prompt you may just press ENTER to leave the editor. To return to the address prompt at any time, use shift/left-arrow. On the right is continuously displayed the current address of the blinking cursor, and below that a "reference address" and the number of bytes (decimal) that the cursor is offset from it. Pressing shift/break will reset the reference address to the current cursor address. The arrow keys move the cursor; shift/up-arrow and shift/down-arrow scroll rapidly. Pressing shift/clear will change the type of dump from hex to ASCII and back again. Under either mode, whatever you type in will be put into memory at the cursor location. A little practice will clarify these instructions. The editor is much easier to use than to explain. It is one of the few utilities in Trakcess that will not return you to the menu via 'M'. You must press shift/left-arrow and then ENTER.

10. (B)uild Format Track

This powerful command allows you to specify interactively a track to be created. There are no restrictions other than overall length on the track contents. You may specify up to 80 different sectors, of differing lengths and types, with any names. You are allowed to change specifications on any sector; when finished, the format track may be created (at DC00H) and (using P) written to any track of the disk. This command is very useful in creating disks that can more efficiently store and retrieve the types of data you wish to work with. For example, you might create a track with only one sector of 3024 bytes. This entire block could be read into RAM quickly using a short sector read routine as explained in Barden's book. You might also use (B) to hand-tailor a track to match one on a "protected" diskette.

To specify a sector you must supply a TN, SN, SL, and must indicate (I)BM or (N)BM type. A typical entry would be:

01,2D,03,I

This specifies a sector with TN equal to 01, SN equal to 2D, SL equal to 03, and of IBM type. Correlating the IBM type and the SL value indicates that the sector will be 1024 bytes long. The next sector might be specified as:

C1,77,42,N

resulting in a sector with TN equaling C1, SN being equal to 77, SL equal to 42, and not of IBM type. Since it is an NBM sector, its length is calculated by multiplying the SL value (which is in hex) by 16 bytes. This sector would be 1056 bytes long. These two sectors may or may not be the only ones on this track.

Once you have created the format track at DC00H, you may wish to edit it directly in memory before writing it to disk with the (P) command. Do this by returning to the menu and using the (E)dit function to work at DC00H. One reason you might do this is to zero out the DAM for a short (say 16 byte) sector that you have created, thus generating a "false sector ID"; that is, one with no data after it. Or you might wish to take out the F7 bytes that are put into the format track after every ID pack and sector record. This F7 is what causes the FDC to write two CRC bytes to the disk when it is encountered. Taking it out will result in a false ID pack (if you take out the ID CRC) or an always-bad sector (if you take out the sector CRC). Such unproductive shenanigans will only be necessary if you are breaking a "protected" diskette.

The (B) command is probably the most fun thing Trakcess offers, next to stepping the head back and forth.

11. (F)igure CRC's

This command will allow you to calculate the two CRC bytes for any block of code in memory, or for any bytes you type in. This is useful when examining damaged or tricky sectors or sector ID's. The CRC algorithm used by the FDC is the IBM SDLC standard. It is sufficient to regard the CRC as a two-byte checksum f the data it's applied to -- a checksum that is always initialized to FFFF. The FDC starts calculating a CRC whenever an Address Mark of any kind (F8 - FE) is encountered during a track write, and whenever a sector is being written or read. A subsequent F7 (on a track write), or the end of the sector, cause the FDC to spit out the two current CRC bytes. On any write, these go to the disk. On a sector read, they are reported back to the CPU, along with an indication of their correctness (as compared to the CRC bytes on the disk). Consult the references for more details on these sequences.

12. (H)ex dump to the printer

This actually gives you a choice of hex or ASCII dump, starting at any address. Like all the other printer options in Trakcess it will only be offered if you have said that a printer is ready. It will pause when the space bar is pressed, and break to the menu if 'M' is pressed.

GENERAL

Trakcess has a lot of error-checking built into it on operator inputs, so you should never accidentally crash it. Also, most requests for input have default values specified -- when in doubt, press ENTER! Remember that entering 'M' will usually take you back to the menu.

One common question about Trakcess is why it will not 'page' through sectors a la Superzap. As mentioned earlier, this is because Trakcess does not assume anything about the next sector on a track. There may not even BE a next sector! A paging feature is not impossible to add, but it is awkward.

Other possible future additions to Trakcess are a Disk Compare and a Disk Search facility. Also, improvements will be needed to keep up with the state-of-the-art in "protected" disks. On the day of its release, Trakcess would automatically analyze and duplicate ANY TRS-80 disk on the market -- with one exception. That is surely no longer true. Contact TAS for information on their upgrade/bug-fix policy for Trakcess, and consider a subscription to their magazine/newsletter. It offers a good deal of value for the money, and may in the future contain articles discussing the uses and finer points of Trakcess.

DISK STRUCTURE AND FORMATTIN

The characteristics and capabilities of the 1771 Floppy Disk Controller completely determine the track structure of any TRS-80 disk. The data sheets mentioned earlier fully describe these capabilities, and Barden's book explains the overall disk layout and the integration of the disk drive and controller with the TRS-80 CPU. What follows here will cover the middle ground. There are some things in the data sheets that do not apply to the TRS-80, and there are many things that can be done that are not discussed in Barden's book (since his book addresses only standard TRSDOS formatting).

The actual number of tracks on your disks is determined by the capabilities of your disk drive, not by the FDC, and is irrelevant here. As detailed above, you can make Trakcess handle any number of tracks. What is important is the contents of each track.

For a track to have any easily-accessible, reliable information on it, it must be set up to hold that information in one or more sectors. A "sector" is an area on the track that consists of an ID block, followed by some filler bytes, followed by a data block. There are also filler bytes before and after the sector area. In general, these filler bytes are unimportant FF's and 00's; it is merely necessary for them to be present. There may be up to 80 or more sectors on a track, and each one may have a different length data block. The ID block of each sector contains four bytes that fairly completely identify the sector. These are referred to in the literature as "track number", "side number", "sector number" and "sector length". I will refer to these from now on as TN, SD, SN, and SL, respectively. Each of these bytes may have any value from 00 to FF, except for the restricted values F7 to FE. It is important to note that REGARDLESS of what disk track a sector is actually, physically on, it may have any of the allowable values for TN, SN, SD and SL. Do not confuse TN with the number of the track that the sector is on! Do not confuse SN with the position on that track of the sector! The first sector on track three could very well have TN equaling C2, and SN equaling 1A. There is not necessarily any correlation. It so happens that standard TRS-80 disks DO make the TN equal to the actual track number, but that is merely a convention.

The value of SD is quite unimportant. It is difficult to even read SD, and harder still to write it. It is always set to 00 as recommended in the data sheet. Trakcess does not presently allow you to specify any value other than 00 for SD. Forget about it.

That leaves TN, SN and SL as the important sector specifications that appear in the ID block. The two bytes TN and SN essentially compromise the "name" of the sector. No two sectors on a track may have the same name (the same TN, SN pair); if they do, the data in the second of these sectors will not be accessible. Aside from their function as a name, TN and SN mean nothing.

The length of the sector's data block, hereafter referred to as the sector's length, is partly specified by SL. The value of SL means two different things depending on whether the sector is IBM type or non-IBM (hereafter called NBM) type. The type of the sector is NOT given in its ID block! To successfully read or write the sector, you must know in advance whether it is an IBM type or NBM type. (Here again is a shorthand term: "reading a sector" actually means reading the bytes of data in its data block.)

The length of an IBM sector is either 128, 256, 512 or 1024 (decimal) bytes, corresponding to an SL value of 00, 01, 02 or 03. These are the only four allowable values for the SL of an IBM sector.

For an NBM sector, the value of SL is multiplied by 16 (decimal) to find the length. A mini-disk track can only hold about 3120 (decimal) bytes, so you will not see sector lengths beyond this.

Standard TRS-80 disks have, as you will find if you do a (S)can on one, ten IBM sectors on each track. Each sector is 256 bytes (SL equaling 01).

The ID block of every sector also contains, as its first byte, an FE. This immediately precedes the TN. This FE is called an ID address mark, or IDAM. It and the four bytes that follow it (TN, SD, SN, SL) are referred to as the ID pack. These are followed by two CRC bytes as described above under the (F) command. These CRC bytes serve as a checksum on the ID pack bytes. The IDAM alerts the disk controller that a sector is coming up. The FDC reads in and verifies (by checking the CRC bytes) the rest of the ID pack, thus determing the sector's name. If this is the sector it has been told to read or write, the FDC then calculates the sector length from the SL value and from its knowledge of the sector type. This done, the FDC awaits the appearance of the Data Address Mark, or DAM. The DAM byte may be any of the four values F8-FB. The TRS-80 standard value is FB, except for DOS directory tracks which get an FA. Every sector data block, or "record", must be preceeded by a DAM to be recognized. The record must be followed by two CRC bytes, which here serve as a checksum on the DAM and the data in the data block. On a sector read, knowing the length of the record, the FDC now reads it in, all the while computing the CRC bytes which it finally checks against those at the end of the record. The FDC will return various error messages if it encounters bad ID pack CRC bytes, a missing DAM, or bad sector record CRC bytes. On a sector write the DAM must be specified. Again knowing the length, the FDC writes the specified DAM out over the existing one, and then writes out the correct number of data bytes to fill the record. All the while it is computing the record's CRC bytes, which it finally writes out over the existing two CRC bytes at the end of the record. Here too the FDC will return various error messages if required. These error messages are returned as codes in the FDC status register, and are detailed in the references. Trakcess will interpret the most common errors for you; if it can't make sense of the error encountered, it reports the code that was returned.

The above description pretty much covers the makeup of each individual sector on a track, the characteristics you may find, and the reasons for them. The (B) command of Trakcess may be used to create a few tracks, and the (R) and (S) commands will read and write sectors on them. Remember that the (B) command creates tracks with "empty" sector records, filled only with E5's. You must write this "format" track off to disk, and then fill each sector with sector writes. Experimentation with these commands should answer most questions you may have concerning track structure. The references, especially the FDC data sheet, do contain almost anything you could wish to know; finding it in there is the only problem.

PERSONAL OPINION

The following are my own thoughts, and do not in any way reflect the position of The Alternate Source. They distribute software by many other authors whose opinions are totally different from mine.

Trakcess, like all other disk access and backup utilities, is sort of a gun. How you use it is your business. It is, however, the biggest gun yet, and as such will cause some concern among those who do not wish others to examine, modify or copy their software.

With that in mind, I have a few words about "protected" software. I don't like it, because it always makes modifying or moving the code difficult, and it is usually a pain to backup and use. People do it because they think that their sales will be hurt by copies being passed around. I say that with 300,000 (or more) TRS-80's out there, and the fact that relatively few owners are in close contact with large numbers of other users, and the fact that most owners are quite willing to spend SOME money on software, there will be plenty of sales potential even after all the swapping. And I challenge those same vendors to say right in their ads that "This software CAN'T be backed up, so don't expect to!". Let them forego all sales to groups of people who try to share the cost of software. Let them take the bad with the good. A vendor has every right to "protect" his software -- as long as he's honest about it. I am proud to say that of the hundreds of hours I've spent on Trakcess so far, not one second has been wasted on making it hard to copy. I hope that Trakcess will eventually reach everyone who wants it. This will ONLY happen if TAS is able to continue marketing it. The best course of action is: when you find software that you genuinely think is good and a good value, TALK IT UP. Tell your friends and contacts. Even more importantly, write in to magazines. The immense size of the market out there will take care of the rest, and everyone will be better off for it.

AARRRGGGHH!!

Should you have any problems loading or executing Trakcess, or have suggestions or comments, contact THE ALTERNATE SOURCE, 1806 Ada Street, Lansing, MI, 48910 by mail, or phone area code 517 either 487-3358 or 485-0344.

----- LIMITED WARRANTY ----------

We have, by careful review and testing, attempted to ensure that this program works as specified in this manual. Nevertheless, there may be errors remaining in this program or its documentation. We would appreciate receiving notice of any errors you may find, and you may receive an upgrade of the latest version available at any time and at no charge by sending a diskette with a postpaid mailer to THE ALTERNATE SOURCE, 1806 Ada Street, Lansing, MI 48910.

Neither the author of this program nor the distributors of this program and documentation shall be liable for any damages which might be caused by any errors or omissions in this product. Should the program be unreadable or damaged, the distributor will replace it upon return of the defective product within 90 days of the date of delivery. There are no other warranties, expressed or implied, including the warranties of merchantability or fitness for a particular purpose.

Copyright (c) 1980 by:

THE ALTERNATE SOURCE
1806 Ada Street
Lansing, MI 48910
Ph. 517/485-0344

and

Roxton Baker