TRSTIMES

Keeping Models 3 & 4 Alive

Volume 1. No. 5. - September 1988 - $3.00

LITTLE ORPHAN EIGHTY

The San Diego trip was wonderful. No smog, no ringing telephones, instead sunshine, fresh air and lots of baseball. The scores could have been better, but everything considered, it was a relaxing and fun few days. I enjoyed it very much as did Alan and Steven, my two VERY expensive kids. I didn't know anyone could eat that many hot dogs in such a short span of time!

As mentioned in the CLOSE # column of last issue, it was time for some serious evaluations on the future of TRSTimes. One thing I learned some years ago, the hard way, is that no one will ever do anything particularly worthwhile without the support of their family. Hence, my first concern was that the constant demand on my time by TRSTimes might interfere in normal family life. When starting TRSTimes in early November of last year, I was concerned that, at this time, I might be burned out and would have nothing more to offer. Also, of course, then I didn't know how well the magazine would be received, nor did I know if it would be selfsustaining or if I would have to dig into the Wolstrup savings account to pay the bills.

Well, my worries turned out to be nonexistant.

My family still thinks TRSTimes is fun and are participating actively; the kids help put the pages in order as well as punching in the final staple; my wife has acted as the in-house proof reader and has, more times than not, edited my rambling thoughts into readable articles. She is a writer, with books and many magazine articles to her credit, so her help has been invaluable.

The support of the TRS-80 community has been fantastic. The subscriptions came in (and still are) at a rate where the initial startup loan has been repaid, all bills are current, enough money to produce and mail out issue #6 is set aside, and there are still a few bucks left over.

I realize that the accountants at 80 Micro or any of the other 'big time' publications, past or present, would hardly classify TRSTimes as a financial success. It is, nevertheless, very satisfiying that when they abort for monetary reasons, a publication such as ours, can exist without red ink on the books.

The response from the readers has also been very encouraging. Article submissions have reached a point where both issues 4 & 5 could easily have been 40 pages instead of 30. However, postal rates, along with some other factors, force the page limit to remain at 30, at least for now.

As you can probably tell from reading the above paragraphs, my fear of burn-out was unneccessary. I am as enthused about TRSTimes, and the TRS-80, as ever before. I feel there are many more things to explore, more new things to learn and much more information to get out. So the bottom line is this:

TRSTimes WILL continue in 1989.

I very much look forward to another year TRSTimes. Plans for exciting new features are on the drawing board. The January issue will see the beginning of a series of Assembly Language tutorials that will be unlike any of the books or articles you threw away because you did not understand. This series will be written in such a manner that almost anyone with a fair knowledge of Basic should see the light and thus get started on the intricacies of Z-80 assembler.

Also planned, due to popular demand, is regular coverage of the machine that started all the TRS-80 fuss, the Model I. We have promises of articles, as well as an in-house working Model I with which we can verify, or if need be, produce material.

There is also the possibility of a series of tutorials on Basic. Letters have indicated that many of you would like to learn how to program. We may just do that.

While other new things are being talked about, be assured that your favorite columns, such as Tim's and Roy's will continue, as will program listings and the exploration of the various DOS'es.

The downside to continuing in 1989 is the fact that we are forced to raise the subscription rate. One of the major reasons that TRSTimes is financially solvent is George Templing. George is the manager of the local print shop, and for the past year he has been kind enough to trade reproducing costs for private MS-DOS lessons (seems only fair that MS-DOS should be the vehicle to promote TRSDOS, doesn't it?). However, this deal is not possible in 1989, so expenses are going up dramatically. Thus it is neccessary to raise the subsription rate to the following:

U.S. & Canada: $18.00 for 6 issues in 1989.
Anywhere else: $23.00 for 6 issues in 1989.
(U.S. funds ONLY, please.)

Re-subscribe now. Make sure your TRS-80 has continued information in 1989. The address is:

TRSTimes
20311 Sherman Way #221
Canoga Park, CA. 91306

Enough of the sales pitch; let's get on with the show:

Welcome to TRSTimes #5

TRSTimes - Volume 1. No. 5. - September 1988

CONTENTS:

LITTLE ORPHAN EIGHTY 2

THE MAIL ROOM 4

CP/M - THE ALTERNATE DOS FOR MODEL 4 6

LIVING WITH A COMPUTER JUNKIE 9

ENHANCING SCRIPSIT 10

OLD COMPUTERS NEVER DIE 13

TRSDOS 1.3. CORNER 16

MACROKEY 19

USING SYS13/SYS 21

TIMS'S PD EXPRESS - HIGH REZ 80 23

JACK'S TIP SHEET 26

DATE YOUR TRSDOS 6.2. THROUGH 1999 27

ITEMS OF INTEREST 28

CLOSE #5 30

TRSTimes is published bi-monthly by TRSTimes publications.
20311 Sherman Way #221. Canoga Park, CA. 91306. U.S.A.

Entire contents [c] 1988 by TRSTimes publications.
No part of this publication may be reprinted or reproduced by any means without the prior written permission from the publishers. All rights reserved.

1988 subscription rates (6 issues):
United States and Canada: $15.00 (U.S.)
All other countries: $20.00 (U.S.)

THE MAIL ROOM

HELP

I recently purchased a computer question bank management system to help me with my increasing classroom workload.

To date I have been unable to locate a source of multi-choice questions (electricity, electronics, mathematics, and physics) in Canada.

Apart from the do-it-yourself route, do you know of a source for test questions in my areas of interest?

Fred W. Happy
RR 3. Box 173
Kingston, Ontario
Canada K7L 4V2

I have just subscribed to your mag thru' NATGUG, and have issue 3 but expect earlier editions in short order. It is good to see someone still cares for us Model III users (I got 2, in case one wears out).

I have a problem and I hope TRSTimes and its readers will be able to help:

One of my Model IIIs came with the OMIKRON MAPPER III CP/M board. version 1.0, 35 track, S/D. OMIKRON (of Berkeley, Ca.) never did send a D/D update. Now out of business, I think.

Does anyone have access to the D/D update and know how I can acquire same? Also I have COMMX, TARBELL, ELECTRIC WEBSTER, & WORDSTAR - all in D/D, with no manuals. It's a shame to let all that software and facility go to waste.

CAN ANYONE HELP - - PLEASE ??

Tony Gerrard
11 Forge Valley Way
Poolhouse Farm
Wombourne Staffs
WV5 8JR
England

Can anyone out there in TRS-80 land help Fred and Tony? Please write them with any information you may have at the addresses given above.

Ed.

KIND WORDS

These days I am breathing a great sigh of relief with the following and momentum your publication and TRS Link are enjoying. I first learned about TRSTimes through TRS Link on the Genie system.

I felt I was being forced to give up my Model IV for an MS-DOS setup although the IV has done everything I needed it to do. Now with the efforts of yourselves and others I can continue to learn and grow in my computing skills.

Thank you for all your efforts and support and rest assured I will support your publication and all others that keep me up and running.

Keep up the good work.

David W. Ireland
Devoted TRS80 User
Key Largo, FL.

I appreciate your efforts and the product of TRSTimes. With issue 1.3. (May) just received I can't express too greatly our THANKS for continuing to support a Great Computer. Please keep me on the mailing list for current issues, Disks, etc.

I am Past President and continuing "Membership Chairman" of the "DALLAS MICROCOMPUTER USERS GROUP, INC". Our common name is DALTRUG-Tandy Users. We have a large SIG supporting MOD I/III/IV/4P as well as the Models II/12/16, and have been in existance since 1978.

Currently we meet each 2nd Saturday, 9:00 am to 4 pm at the Dallas INFOMART, 1950 Stemmons Frwy, Dallas, TX. Our day covers such SIGS as "STARTEXT USERS, COLOR COMPUTER, TANDY 1000,2000,3000, M1/III,IV,4P, MAIN BUSINESS MEETING, MODEL II/12/16/PUB DOMAIN SOFTWARE, AOS-9 USERS, LAPTOP COMPUTERS, PERSONAL ANCESTRAL FILE, AND from time to time SIGS on Basic and 'C'". A very active and Tandy oriented group.

We maintain one of the better BBS's in the area with plenty of software for the TRSDOS machines, as well as MS-DOS. 24 hours with 300/1200/2400 on FIDO. An open board at (214) 234-4952.

Please call on us if we can be of help.

Fred A. Driscoll, Jr.
Dallas, TX.

We thank both David and Fred for the kind words. It is this kind of support and enthusiasm that will keep the TRS-80 alive for a long time to come.

Had 80 Micro only realized how much we TRS-80 types love the machine.....

Ed.

MISSING PERSONS

EASYDOS

Can anyone help me get in touch with Robert Ackerman, the author of EasyDos, a DOS shell program for the Model 4? I liked it and sent him my registration in April, but have never heard from him. My check has not been cashed, and his phone has been disconnected (I even sent a label and return postage to return my check in case he no longer supports the program). I'd like to find out what has happened.

Don Singer
3726 Skyline Dr.
Scottsbluff, NE. 69361

The latest address we have for Robert is:

ROBERT ACKERMAN
906 SARTORI AVE. #2
TORRANCE, CA. 90501

Ed.

P2DOT

Several readers were intrigued by Steven Jerkins' review of P2DOT in Issue 4 and would like to get in touch with the author of the program. We did some research and, hopefully, this is the author's current address:

PAUL F. BARNETT
BOX 170
HENDERSON, KY. 42420

Ed.

ASSEMBLY LANGUAGE TUTORIAL

Please continue in 1989 so I can reap the benefits of the promised tutorial on assembly language. I have tried to read through several books and articles on the subject, even took a night class, but I get lost almost immediately. I am a fair Basic programmer, but I just can't grasp the concept of assembly language. Will your tutorial provide me with some understanding?

Brian Colson
Jacksonville, FL.

First, as stated in Little Orphan Eighty, TRSTimes has committed to continue with another six bi-monthly issues in 1989. This means that, among other things, the assembly language tutorial is on the drawing board. The plan is to aim this tutorial at people, just like you, that has a fair- to good knowledge of Basic. As much as possible, we will treat assembly language as if it was Basic itself.

Teach you assembly?
NO, only YOU can do that.
Making you see the light to get you started?
YES, I'd say there's a good chance.

Ed.

CHANGO80 CORRECTION & SOLUTION

I have invested several hours of my life typing in, finding and correcting my typos, and playing CHANGO80 from issue 4. I have found the game very enjoyable, challenging, but mostly frustrating: I can't get a perfect score. What is the correct sequence?
P.S. I resent the comment in C$.

E. W. Simmons
Wausau, WI

Even though the listing for CHANGO80 contains two offsetting minor errors, they in no way prevent you from getting a perfect score at 169 moves. However, a perfect score does really take 170 moves, but since the logic in line 500 does not increment the score if the puzzle is solved, a perfect score is indicated at 169 moves. To correct this change lines 500 and 510 to read:

```
500 M=M+1:IF B=0 THEN GOSUB 30:
GOTO 510 ELSE GOSUB 30:GOTO 320
510 PRINT@256,ER$:IF M=170 THEN
PRINT@285,"Perfect score - GENIUS":GOTO 600
```

C$ was intended to be tongue-in-cheek. By all means remove it or change it to whatever you would like it to be.

CHANGO80 is meant to be challenging and, YES, also frustrating. This type of puzzle-game loses its charm once you know the answer and can solve it at will. However, since several letters have indicated the same frustration, we will give the sequence for a perfect score. Use it once to get the idea, and then promise to never look at this solution again.

```
7,8,5,8,7,8,6,8,7,8,
3,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8,
4,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8,
1,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8,
4,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8,
3,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8,
4,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8,
2,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8,
4,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8,
3,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8,
4,8,7,8,6,8,7,8,5,8,7,8,6,8,7,8
```

Ed.

CP/M - The alternate DOS for Model 4

by Roy Beck

A reader has asked me to explain SUBMIT and XSUB. That's easy in a simplistic sense: SUBMIT is the analog of the DO (JCL) files of TRSDOS, and XSUB adds some capability. But that's hardly a satisfactory explanation, especially if you do not understand DO files. So let's give it a whirl.

A JCL (Job Control Language) file in any DOS is a means to automate a repetitive task. Obvious examples at the PC level are, initiating a terminal program which requires setting a group of parameters to certain values each time, such as entering name and password information; and, for me, attempting to initiate my dBaseII program with the files necessary to do the monthly bookkeeping for the club of which I am the treasurer. Since these two tasks are under different DOSes, it may be helpful to explain both of them.

Of course, there are many different terminal programs around, but the ones I use to move files from my Mod 100 to my Mod 4 under TRSDOS are the ones I happen to use. The TERM program is in ROM in the Mod 100, and the corresponding Mod 4 program COMM is an integral part of TRSDOS 6.X. (There is a corresponding program named LCOMM in LDOS 5.X). The Mod 100 requires XON/XOFF handshaking at anything above 600 baud, so in order to achieve reasonable speed, I run at 9600 baud when moving files from one to the other via my null modem.

Setting up the Mod 100 is easy, as the menu shows me the few commands necessary, and I can remember them easily. But the TRSDOS commands are another story. It's true, there are only 3 commands there to initialize, but they are not obvious, and I inevitably forget something, put a comma or space in the wrong place, or do something else which requires me to retype the command (and curse Roy S. under my breath). I have found out it is much simpler to write a little DO file for this task, and get on with the job at hand. (As the old saying goes, "When you are up to your ass in alligators, it is sometimes difficult to keep in mind that your original goal was to drain the swamp!"). Since my goal is to transfer files from the Mod 100 to the Mod 4, anything which minimizes the alligator headcount and facilitates the original goal is welcome.

You may wonder why I don't write on the Mod 4 under Allwrite in the first place. The answer is easy. The Mod 100 is so much more PORTABLE! (I am writing this article while soaking in my hot tub). Since the Mod 100 is battery powered, and all its voltages are below the personnel safety limit of 50 volts, I can safely use the Mod 100 in a wet environment. Don't try that with any computer that requires 110 VAC for operation!

The TRS DO file (MOD100/JCL) which initializes the terminal program is as follows:

```
SET *CL TO COM/DVR
SETCOM(W=8,P=N,B=9600)
COMM *CL
```

To execute the file I simply type **DO MOD100**, and almost immediately I am ready to begin file transfers.

TRSDOS has a BUILD command to simplify the creation of DO files. When you use this, you call "BUILD filename" with no extension. Build automatically adds the /JCL extension for you. Since CR's are legitimate entries in DO files, you cannot terminate creation of DO files under BUILD with a CR. Instead, use **SH + CTL@**, which completes the construction of the DO file and returns you to DOS READY.

The JCL language allows the inclusion of various comments which will be printed to the screen during execution to guide and inform the operator. The line will appear upon the screen, allowing instructions to the operator to be included, such as: "Don't forget your wedding anniversary on June 10th!" A Pause command is available to suspend operation while the operator changes disks, etc.

SUBMIT, under CP/M, allows you to do almost anything automatically from a SUB file that you could do by typing on the keyboard at the DOS READY. In fact that is what occurs when using SUBMIT. The keyboard driver is temporarily replaced by a special driver which feeds in a script of commands from a buffer, supplying a command whenever DOS (CP/M) needs further keyboard input. Of course, a DOS error or a faulty command in the SUB file will lead to trouble, as CP/M is not the most forgiving of DOSes. But good equipment and careful preparation of SUB files will allow you that "Ain't I Smart?" feeling when you get it working.

My attempt to automate dBase was unsuccessful, but the process is still instructional. Following was my approach. In Montezuma Micro's CP/M, the command to start up the dBaseII program and call the club files would have been **SUBMIT VCC**, which should have begun execution of the file named VCC.SUB. (The name of every CP/M SUBMIT file

must end in .SUB). The listing of the SUB file was as follows:

```
XSUB <CR>
DB <CR>
<CR>
USE B:VCCFILE INDEX B:NAMELIST <CR>
```

SUB.COM is a transient command and therefore must be available on the logged on disk.

The XSUB in the SUB file stands for eXtended SUBmit, and allows SUB to continue functioning inside your application program after it has loaded and begun execution. SUB by itself cannot do this. (TRSDOS also cannot do this). By the way, not all application programs will accept the tender ministrations of XSUB. The only way to be sure is to try it yourself to see if your application program is compatible. Waite and Angermeyer, authors of the "CP/M Bible", say XSUB will work with most programs, except those which are "ill-behaved". This means the programs must use only official DOS calls, instead of taking "shortcuts". The only way to know if your program is "well-behaved" is to try XSUB and find out.

To use the XSUB feature, it is necessary to place the command XSUB on a line prior to calling of the program within which XSUB is expected to function. While not mandatory, as a general rule you should place XSUB on the first line of the SUB file. Once called, XSUB remains resident. If its services are needed for a second program in a long SUB file, it does not need to be called a second time. Note that XSUB.COM, (which is a transient command), must also be available on the logged on disk.

In my case, the SUBMIT command worked as far as loading and beginning execution of dBaseII, but dBase thenceforth ignored the SUB file, and did not accept the remainder of the commands. Evidently dBase sets up and uses its own keyboard driver and ignores CP/M's driver. It was interesting to note that, after QUITting dBase, SUBmit resumed activity and issued the remainder of the command list to DOS, which had no idea what to do with them!

We now know dBaseII does not accept XSUB, but only testing will tell about other programs. To paraphrase the old TV commercial, "Try it, you may like it!".

By the way, CP/M does not contain an equivalent of TRSDOS' BUILD command. To create the SUBMIT file, you must use an editor or word processor which creates a straight ASCII file. By this I mean a file with no unprintable graphics characters in it. ED.COM, which comes with CP/M, is perfectly capable of doing this, although many aspersions are cast in its direction because it is not a full-featured editor. However, it does small tasks like creating SUBMIT files easily. Another popular method is to use WordStar in the non-document mode. (The document mode adds control and graphic characters to the file for formatting purposes, which is unacceptable to SUBMIT).

SUB also allows comments in a SUBmit file. When the file is created, comments may be inserted, which will appear upon the screen when the file is SUBMITted. Each comment must begin on its own line, prefixed with a semicolon (;).

Another feature of the SUB command is the capability of accepting variables to be replaced with real values at the time of execution. Variables in the file are represented by $0 through $9. $0 is a special case. It will be replaced with the name of the SUBMITted file; $1 through $9 will be replaced by their respective parameters on the command line.

The following example is taken from Waite and Angermeyer. The example initiates a new system disk and is actually inappropriate for the Mod 4, as Monty (of Montezuma Micro) has provided a simple backup command as a part of our CP/M package. Nevertheless, I will present the DISKINIT.SUB routine for explanatory purposes.

DISKINIT.SUB

```
FORMAT
SYSGEN
;Copy all COM files from Drive $1 to $2
PIP $2:=$1:*.COM(V)
;Display directory of Drive $2
DIR $2:
```

This routine will format a blank disk, sysgen the system onto it, copy all .COM files from drive $1 to $2, and finally display a directory of the new disk.

To execute this routine, one would use the following command:

```
SUBMIT DISKINIT B D
```

Wherever $1 appears, B would replace it; similarly for $2. The new disk would be in drive D, and the .COM files would be copied from drive B. Where the number of parameters fails to match the number of variables, CP/M will follow certain rules. If there are too many variable values in the parameter string, the extra values will be ignored. If there are too few, the corresponding variables will be replaced with nulls. All of this will be without notice to the operator. Needless to say, the results may not be as you expected. As always with CP/M, BE CAREFUL!

A $ in a comment line will also be treated as a variable to be replaced by something from the parameter string. But you may well wish to have a real $ appear in a comment line. In this case, put two

(2) $'s in the comment line ($$). This will result in one being displayed and the other one being dropped into the bit bucket.

In the course of its execution, the SUBMIT command will always create a temporary file on your logged on disk named $$$.SUB. This file incorporates the variable substitutions discussed above, and is created immediately after the SUBMIT command is started. For this reason the logged on disk must not be write protected, and must have sufficient space for the $$$.SUB file, which will usually be the same size as the original .SUB file.

There are some known bugs in SUBMIT which will never be fixed, but can be partially worked around. The CP/M manual by Cortesi, "Inside CP/M" (included with Monte's CP/M package) points out these bugs. Cortesi mentions them on page 130. Briefly, SUBMIT files can, and often do contain control characters, such as ^Z, as is used in PIP and ED to terminate them. SUBMIT is supposed to accept either ^Z or ^z, replacing either one with the control z value **1AH**. The bug is that SUBMIT won't accept ^Z, although it will accept ^z. The workaround for this is discussed on pages 192 and 366 of the Cortesi manual.

According to Cortesi, there is another bug in the SUBMIT command (P. 366). It won't accept a command line of zero length, preferring to crash instead. There is no fix for this problem. You will note that I created such a problem in my own attempt, where I attempted to bypass a date entry requirement with a <CR>, but it did not bite me. It never got that far in the execution! I am sure it would have caused trouble if XSUB had been allowed to proceed, but unfortunately, dBase defeated me.

Another reader of these epistles to the TRS faithful has taken me (mildly) to task for an omission in my essay on PIP. He points out that I omitted mention of a parameter for PIP which can be most useful when needed. Mea Culpa!

The missing parameter is Z. Z is a parameter which will zero bit 7 of any byte of a file being transferred by PIP. The format would be:

A>**PIP B:filespec A:filespec (Z)**

The need for (and utility of) this parameter arises when you wish to print a (document mode) Wordstar file directly from DOS or when you wish to load a W* file into some word processor which chokes on W*'s formatting characters, which include bytes with bit 7 set. If you try to print a formatted file under DOS, the printer will perform unpredictably, depending upon how it responds to control and graphics codes. My own printer will change fonts, go in and out of compressed and expanded modes, etc. Very disconcerting. Of course, the need is to get rid of the graphics characters, and PIP can do this by zeroing bit 7 of every byte. This does not really get rid of the offending bytes, it converts them to characters in the range of 00 to 7FH, which makes most of them printable characters. They will now print as ordinary letters, but at least the printer can handle them. Of course, any which are converted to values in the range of 00 to 1FH can still cause trouble, causing unexpected form feeds, etc. The unexpected form feeds can in turn be eliminated by adding yet another parameter, F, which gets rid of all form feeds. The sequence of parameters is important, and in this case must be (ZF). If the order is reversed, only form feeds originally in the file will be eliminated.

By first PIPing the formatted file to a new file with the graphics bit unset, you can load the file into whatever WP you have and remove the offending characters. Then it can be printed directly with no trouble.

PUBLIC DOMAIN PROGRAMS

NEW PROGRAMS

from the Valley TRS-80 Hackers' Group public domain library for Model I, III & 4

Send SASE for annotated list

Sample disk $5.00 (US)

VTHG

BOX 9747

NORTH HOLLYWOOD, CA. 91609

LIVING WITH A COMPUTER JUNKIE

by Barbara A. Beck

Ten years ago my husband, Roy, walked into a Radio Shack and bought a computer. His first.
Life hasn't been the same since.

It was a Radio Shack Model 1 and when he got it home he found a perfect place for it in his home office. He was happy. I was happy he was happy. I thought that would be the end of it.

The Model 1 kept Roy entertained whenever I had to be out of the house helping my parents, who were both in their late 80s. When I'd get home I knew I could always find him in his office, hunched over his machine like a drinker at his favorite bar. Sometimes he'd try to get me to join him, but having no predisposition towards computer addiction I'd only sit down to learn a program or two, like Electric Pencil or a genealogy program, but that was it. I never got hooked. I could stop any time I wanted. It might be days before I'd need to use a computer again. And of course I never computed alone. I guess you could say I'm strictly a social computer user. (I was also afraid I'd destroy something.) But Roy was different. Once he got started computing he couldn't stop.

His behavior began to change. He started hanging out with the boys at the TRS 80 computer club down in Orange County. And we lived in Los Angeles! But Roy would go to any lengths to be with his junkie companions, so once a month he'd spend all day Sunday commuting back and forth to his meeting.

One day I came home and there was a strange new computer, a Model 4, sitting in the living room. "What's that?" I asked Roy. "That's **ours**", he said--and immediately took it into his office with him and shut the door.

Next there was a printer. Then a hard disk. And then a printer upgrade. The machines outgrew his office and began creeping into the livingroom and the patio where they clung territorially to various pieces of furniture. Computer newspapers and books spread themselves over every available flat surface in the living room and when we put up a new book shelf, they overtook that too. Next we put two sheets of plywood across part of the open beam ceiling in the patio to create more storage space, and in no time books and magazines invaded that area--and won. Cleaning became a chore--until I discovered the feather duster. But I never did figure out how to polish the wood.

As Roy's condition progressed, I knew that there must be a self-help group for the loved-ones of computer junkies to go to. I found one! It was a Radio Shack class on the Model 1. I signed up. I guess their philosophy was: "If you can't beat 'em, join 'em!" Roy was happy having a computing companion at last. And when I was given my very own computer, a RS 100 laptop portable, I figured at least we'd go down the drain together. But to my horror I returned home one day to discover that Roy's own computers weren't enough for him--he'd borrowed mine!

"Why can't you just **stop** after one or two?" I cried. "Have you no willpower!"

Chagrinned, Roy did try harder after that to share his supply with me. Being a true computer connoisseur he set out to educate me, patiently explaining every facet of each machine, each program's code, each time-saving technique.

We had good times together, there's no doubt about it, but sadly it didn't keep Roy's addiction from progressing to the point where he began staying out at night at least five or six times a month. He said he was going to meetings, but I'd find names like SAGATUG and VTUG and VTHG (probably her sister) written on bits of paper around the house.

When he began going to Computer Swap Meets I decided to try to join in with him one more time. "All right, all right, I'll go to the swap meet with you!" I said. "I'll even buy something computer-related!"

I came home with an antique clock.

Not too long ago I realized it had been a couple of weeks since Roy had come home with a new computer or a new printer upgrade or a new computer book. Could it be that he was changing after all? Was he losing his addiction?

Then I opened up a drawer and found a bunch of computer pieces hidden under a towel.

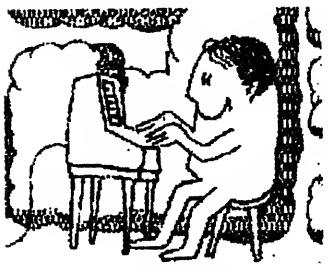
"I assume those are yours," I said to Roy, pointing to the computer parts in the drawer.

"Those are **ours**", he said. "They may be of use some day!"

Once a computer junkie,
always a computer junkie!

ENHANCING SCRIPSIT

by
David Goben



Scripsit modifications can be found almost anywhere. The popularity of the word processor and its ease of use has made it an instant winner. Unfortunately, as time has passed, printers have become more and more powerful, being capable of MUCH more than the printers current to the time of Scripsit's development, making the word processor seem primitive in many respects. Hackers later found out some amazing things about Scripsit as they attempted to patch it to accomodate their new printers -- it was EASY to patch. The structure of Scripsit is so logically defined, and laid out so understandably, that any hacker worth their salt could figure out what it was up to in any section of code in only a short period of time.

Since the popularity of my Shareware MODSCRIP package has begun to grow tremendously, many of its eager users have started asking me how I manage to do so MUCH to a word processor such as SCRIPSIT, and still not take away a single byte of text buffer space. It's easy: I KNOW the system. That is probably the greatest secret to any such 'amazing' feat. Those who are curious enough will explore. My own natural curiosity has brought me a long way. When I did not understand something, I made every effort to do so. Most often this led me to reading hundreds of books, learning new computer languages, or simply stepping back and looking at a single thing from various points of view. Virtually all of these feats apply to how I developed MODSCRIP.

I started with a disassembled listing of Scripsit. I then went through the task of figuring out what each and every routine did. In doing so I found a LOT of free space in the program. Scripsit is FILLED with unused space and un-used code. There is even a look-up table that is totally ignored by the system. After mapping them out and filling them with enhancement routines, I set about to find MORE space by examining the WAY that the original programmer had written Scripsit's code. From that I figured out how to optimize it, to make it shorter. Each routine that I made shorter resulted in more free bytes to play with. I had often spent hours in order to figure out how to free up only one or two bytes. I cannot begin to count the times that I thought I had squeezed the last free byte out of the program, only to wake up the next morning realizing how I could free several dozen more. With each set of free bytes came at least one more enhancement, until the sytem grew to often rival, and sometimes surpass that of SuperSCRIPSIT.

Add to this my worst habit: being a programmer, I can never seem to leave a finished set of code alone. I will always tinker. Often that tinkering results in not only finding enough space to place two enhancements where before only one enhancement existed, but also to make the combined length of the two enhancements shorter than the original block of code. My other bad habit is that I will sometimes totally re-write a program to save only one byte of code. I have this weird thing about making any program I write its ABSOLUTE shortest. These bad habits also lead me to receive a lot of curious stares. For someone will 'wonder' where a certain routine might be located in a huge program, and I will often immediately state the address right off the top of my head. In the case of Scripsit, I probably know the system as well, if not better than the original author. After documenting each and every byte in the system, I guess it kind of burned itself into my memory. These habits are exemplified by the fact that I had developed version 1.0.0 of MODSCRIP in January of 1983, and even now I am STILL hacking away, completing version 9.0.0 in early June of 1988, and only the other day (the last day of June, 1988) I figured out how to apply even more patches to accomodate printers that require a SECOND code after the backspace code, to back up a full character space, such as is needed on the DWP-410.

In early 1977 Beve Woodbury up at 80 MICRO asked me if I knew how to add a couple of enhancements to Scripsit, to answer a Feedback Loop question. In a matter of an hour I had developed a program that applied not only those enhancements, but several others as well. Unfortunately this amazing little program was never published. Only the inquiring reader received a copy. I later decided to present this program in the July 1988 issue of Computer News 80, but being sqeezed for space, I had to put it back on the shelf. When Lance Wolstrup asked me to write an article for TRSTimes, it was like being the fastest draw in the West; I had an instant article ready.

The patches I developed for 80 MICRO have been further enhanced, just today, by patches to make the Model I's SCRIPSIT/LC program to work not only on BOTH the Model I and Model III, but under ANY Model I/III DOS, except for CP/M. With these new enhancements you can print such characters as brackets, underscores, control codes, and codes greater that 127 directly from the keyboard. It will also recognize the RIGHT SHIFT key on the Model III, respect Model I/III high memory settings, exit to DOS rather than do a reboot, print to both the Model I and III printer ports, and work with other non-Radio Shack printers that it before would not operate on. The

modification program itself give you the option to force it to send a line-feed after a carriage return, if your printer requires it, thus eliminating the need for a printer filter for some printers.

To apply these patches you need two things. First you need a copy of the Model I version of Disk Scripsit 1.0, called SCRIPSIT/LC. If you are using a Model III, then you may have to convert SCRIPSIT/LC over to your DOS by using a CONVERT utility, such as CONVERT on TRSDOS 1.3, or REPAIR on LDOS. Second, you need my modification program, called SCRPATCH/BAS. The program listing following this article is just that. NOTICE that you must copy SCRIPSIT/LC to another file called simply S/CMD, which can be done by using the command COPY SCRIPSIT/LC TO S/CMD. S/CMD is what SCRPATCH/BAS will operate on. NEVER work with an original copy. ALWAYS use a backup copy. If errors occur and you destroy your only existing copy, you will have only yourself to blame. NEVER be TOO anxious to do ANYTHING. Always be organized and clear-headed. Those in a rush usually end up taking more time to do something than someone with patience will, anyway.

To apply the modifications, go into BASIC and run the SCRPATCH/BAS program. It will open S/CMD and modify it. You will be asked one question. If you want a Linefeed (LF) to be issued after each Carriage Return (CR), as SOME printers require, then answer the prompt with 'Y'. If you have a Radio Shack printer, or any printer that advances a line each time only a carriage return is issued, then answer 'N'. That's all there is to it. All that is left is to go back to DOS (CMD"S") and enter S, to run S/CMD.

When you are in Scripsit, try this: hold down the '@' key and press the number keys '1' through '9'. How about that? Bet some of you never realized you had these symbols. Next, put yourself in the lower-case mode (if you are not already there) by pressing SHIFT-@. Notice that you can type capital letters by holding down EITHER the left or the right shift keys on the Model III (no great feat on the Model I). If you want, you may wish to load a document and try printing it, to test printer output.

How can you send a control code to the printer? Easy. Hold down the '@' key and press the number zero key. Notice the neat little symbol displayed. Follow this with a code that is a value of 32 HIGHER than the control code. For example, on most Radio Shack and Epson printers, you can turn underlining on with the code 15. 15 plus 32 equals 47, which is the ASCII code for the slash (/). So type a '/' IMMEDIATELY after the new symbol you just created. Turning off underlining would be to send a 14 code out. So after the text to be underlined, again hold the '@' key down and press the zero key, and then press the period key (code 46, which is 14 + 32).

If your printer supports ASCII codes higher than 127, then you can use yet ANOTHER special code. Hold down the '@' key, but this type type the colon (:) key. Here is yet ANOTHER new symbol. This would be followed by a key that has a code that 60 LESS than the desired code. Thus, to print the code 171, you would subtract 60 from the code, which is 111; the ASCII code for the lower case 'o' (OH) character. Thus you would hold down the '@' key and press the colon (:), than then press the 'o' key (you must be in the lower case mode to do this, obviously).

You have just made Scripsit into a much more powerful system. Obviously, this is only a sampling of the features you can add to Scripsit by using my MODSCRIP package, which converts Scripsit to operate on the I/III and 4 (in the 4-mode, with MAJOR new features) or any other, such as Powersoft's PowerSCRIPT, which will modify the I/III or 4 version of Scripsit.

If this has whetted your appetite to enhance your Scripsit system, you can check out:

Powersoft
4951 Airport Parkway, Suite 700
Dallas, TX 75248

They sell PowerSCRIPT for $24.95 (US funds), plus $3 S/H. COD add $5. Foreign add $10 (Texas add 8% sales tax).

Their package will operate on Model I's SCRIPSIT/LC, Model III's SCRIPSIT/CMD, and Model 4's SCRIPSIT/CMD.

You can also obtain my own MODSCRIP package by sending a formatted disk (single-sided, Model III or 4 Double-density formats only, such as LDOS, LSDOS, TRSDOS 1.3, NEWDOS80, MULTIDOS, DOSPLUS, etc) in a return mailer with return postage plus $3 (US) for handling. This insures that you get the latest version. If your BBS operator already has it, be sure they have the version that also contains the files PRINTERS/TXT and PRTRBKUP/BAS. If they don't, then let them know where to get it. If you have Scripsit, but not the Model I version, send proof of purchase of YOUR version, such as an ORIGINAL customer service bulletin, or even your original Scripsit disk (these will be returned), and I'll include the SCRIPSIT/LC version on the MODSCRIP disk . PLEASE be sure to include all required items: formatted disk, return mailer, return postage, and $3.

Good luck with Scripsit, whether you use my package, Powersoft's, the ones presented in this article, or just the original, unmodified version.

David Goben
28 Monticello St
Willimantic, CT 06226-1325

SCRPATCH/BAS

```
10 CLS:PRINT"SCRIPSIT/LC Enhancements, By David Goben"
20 PRINT"-----------------------------------------"
```

```
30 PRINT"Run on backup copy of SCRIPSIT/LC
called"
 40 PRINT"S/CMD.":PRINT:PRINT"Opening
S/CMD..."
50 ON ERROR GOTO 0:
OPEN"I",1,"S/CMD":CLOSE 1
60 OPEN"R",1,"S/CMD":FIELD 1,1 AS A$
70 PT = PEEK(VARPTR(A$) + 1) + 256*PEEK
(VARPTR(A$) + 2)
80 IF PT > 32767 THEN PT = PT-65536!
90 HX$ = "0123456789ABCDEF"
100 PRINT:PRINT"Applying Patches...":
PRINT:GOSUB 160
110 A$ = "":LINE INPUT"Add LF after CR (Y/N)?
";A$
120 IF A$ = ""THEN 110 ELSE A = ASC(A$):IF A96
THEN A = A-32
130 IF A = 78 THEN 150 ELSE IF A < > 89 THEN
110
140 GET 1,1:POKE PT + 25,32:PUT 1,1
150 CLOSE 1:PRINT:PRINT"Modifications Com-
plete.":END
160 GOSUB 190:IF A$ = "END"THEN RETURN
170 RS = A + 1:GET 1,RS:GOSUB
190:PR = PT + A:GOSUB 190:FOR CT = 0 TO A-1
180 GOSUB 190:POKE PR + CT,A:NEXT:PUT
1,RS:GOTO 160
190 READ A$:IF A$ = "END"THEN RETURN
200 A = INSTR(HX$,LEFT$(A$,1))*16 +
INSTR(HX$,RIGHT$(A$,1))-17
210 RETURN

220 'remove initial out of CR at startup
230 DATA 00,48,03,00,00,00

240 'printer output fixes
250 DATA 14,B2,03,CD,00,52
260 DATA 15,7A,03,CD,00,52
270 DATA 29,3B,03,CD,00,52
280 DATA 29,42,03,CD,00,52

290 'allow control code output and
high-bit byte output
300 DATA 00,04,3F,C5,4F,3E,00,B7,20,1C,79,FE,
7E,28,11
310 DATA FE,7F,28,0D,CD,31,52,FE,0D,18,05,0E,
0A,CD,31
320 DATA 52,AF,32,03,52,79,C1,C9,FE,7E,79,20,
04,E6,1F
330 DATA 18,02,C6,60,4F,18,E8,3A,E8,37,CB,7F,
20,F9,79

340 'send byte out to printer
350 DATA 32,E8,37,D3,F8,C9

360 'if you want printer data sent to ROM
(x'003b'), then use the following data line-
instead of the above data line:
370 'DATA D5,CD,3B,00,D1,C9
380 '
390 '

400 'fix a bug in scripsit
410 DATA 29,4B,01,3F

420 'fix right/left shift key check for model III
430 DATA 1E,30,09,3A,80,38,E6,03,C8,3E,01,C9
440 DATA 0F,09,03,CD,B8,6F
450 DATA 0F,35,03,CD,B8,6F

460 'look-up table for special characters
470 DATA 06,67,1C,9B,1B,9C,1C,9D,1F,31,5B,32,
5C,33,5D
480 DATA 34,5E,35,5F,36,60,37,7B,38,7C,39,7D,
30,7E,3A,7F

490 'routine to access special characters
500 DATA 0F,85,18,E5,06,0E,21,4B,58,BE,23,28,
06,23,10
510 DATA F9,E1,18,08,7E,E1,18,20,00,00,00,00

520 'exit to dos rather than reboot
530 DATA 13,E5,02,2D,40

540 'fix printer test
550 DATA 14,97,03,80,FE,00
560 DATA 14,A8,03,80,FE,00

570 'allow high memory to be recog
nized
580 DATA 00,64,0A,CD,E1,77,00,00,00,00,
00,00,00
590 DATA 26,79,0D,2A,11,44,3A,25,01,FE,49,C8,
2A,49,40,C9
600 DATA 06,1C,17,45,4E,48,41,4E,43,45,44,20,
42,59,20
610 DATA 44,41,56,49,44,20,47,4F,42,45,4E
620 'FIX STACK FOR ALL DOS SYSTEMS
630 DATA 00,44,01,E0
640 DATA 00,D4,01,E0
650 DATA 04,33,01,DE
660 DATA 07,01,01,DE
670 DATA 0C,64,01,DE
680 DATA 11,53,01,DE
690 DATA 1C,CE,01,DE
700 DATA 1E,EE,01,DE
710 DATA 28,E6,01,DE
720 DATA END
```

"Old Computers Never Die!"

(Buying and Selling Used Micros)

by Fred Blechman

(C) Fred Blechman 1988

You're ready to move-up to another computer - but what do you do with your old one? If you could just sell your orphan instead of sticking it in the closet you might have a few bucks toward your latest "gotta' have." Or maybe you can breathe new life into an old corpse with some jazzy add-ons. This article will describe how I recently sold four of my old computers and bought a "new" one. I'll also cover the way to get add-ons and backup units at a bargain.

Computer Junkies

There are two kinds of "computer junkies" - the Upgraders" and the "Never-Say-Die" types. The Upgraders buy every new microcomputer - usually as soon as it's announced - no matter how many other machines they already have. I tend to be an Upgrader, which accounts for the long list of machines I've owned: TRS-80 Model I, three TRS-80 Model III (one with CP/M), one TRS-80 Model 4, two TRS-80 Model 4P, TRS-80 MC-10, Coleco ADAM, two Sinclair ZX-81, Sinclair Spectrum, Timex Sinclair 1000, Timex Sinclair 1500, Timex Sinclair 2068, Sinclair QL, Sanyo MBC 555, Apple IIc and IBM PC/XT.

Now, I've had good reasons to own all those machines. In the last nine years I've written almost 300 magazine articles and five books about microcomputers. I had to have these machines to write about them, didn't I? Sure I did! And I had to have as many as two or three of some types just in case one failed, didn't I? Of course!

Well, my rationalization might be different from yours, but if you're at all dedicated to microcomputing as a hobby or a business, by now you have an accumulation of micros, and some of them are just gathering dust. After many, many hours of sweating over a particular keyboard, a relationship develops and you just can't let go. Am I right? It seems that the thought of selling one of your machines is akin to selling one of your children!

Besides, who would want them? (The computers, that is, not your wonderful, obedient children.) The other kind of computer junkies, the Never-Say-Die types, can't give up the micro they started with, ferociously responding to those who would have them put aside their long-obsolete machine for something in today's world. The Never-Say-Die types almost become cultists, embracing every possible add-on to their under-designed, under-powered, under-memoried machine, trying to somehow make a silk purse out of a sow's ear. Never mind the tangled cables, corroded connectors, eye-tiring displays and the like. They join together in user groups devoted to maintain the survival of their TRS-80 Model I, ADAM, Timex, Sanyo or old Atari and Commodore models that have been out of production for years!

The Upgrader/Never-Say-Die Marriage

A great opportunity exists for making both these groups happy. All they need to do is find each other! It has been said that for every item someone wants to sell, there's an eager buyer - if they can only find each other.

Here's what I mean. Let's say you're an Upgrader, and have finally run out of space for your old machines. All you need to do is find a Never-Say-Die type that won't let go of the machines you'd like to sell.

Or you're a Never-Say-Die type that's in love with the Texas Instruments TI-99. You need to find someone that has a TI-99 sitting in a closet.

The Big Decision

I'm going to assume you have one or more micros you have finally decided to sell. I'll lead you through the process and then summarize the situation if you are looking for a used computer.

The first thing you have to do is make the decision to get rid of one of your machines. I'd suggest you offer only one at a time if you have several, since that will help you target your market. You can always have some others on your "next-to-go" list, but concentrate on one at a time.

Gather everything you have on this computer. That means hardware (including cables and peripherals),

software, documentation, and books. You'll probably be surprised at all you have accumulated for this now-dead machine. Remember, however, someone out there would love to have this little beauty, and the more you have to support the machine, the more it's worth to the next user.

What To Charge

Now you need to decide how much to charge for this "conglomeration of obsolete junk" - and that's exactly the way you must think about it! The fastest way to sell your machine is to sell it for a low price. This is not easy. You may remember, painfully, what it cost you - and the many, many hours you labored learning how to use it. This bears almost no relation, however, to what you must charge.

On a national level, the National Association of Computer Dealers (NACD) provides survey information on the prices for over 12,500 new and used microcomputer hardware and software items to Sybex, Inc. (2344 Sixth St., Berkeley, CA 94710), a major microcomputer book publisher. Sybex periodically publishes that NACD information in "The Computer Blue Book", available at your computer book store.

Also, the Boston Computer Exchange (BCE) offers hundreds of used computer accessories and equipment for sale. This information is updated daily on CompuServe, an on-line information and electronic mail service accessed through a modem. (Call 614-457-0802 for a brochure.)

CompuServe subscribers can get on line and inquire about any item listed in the BCE database or leave a request to list their used computer equipment with the BCE. The BoCoEx Index, a weekly report of market prices and analysis of marketing trends for used computer equipment, is also available for a $1 surcharge per access. CompuServe users simply type GO BCE at any system prompt.

At the local level, check the classified ads in local papers (such as the "PennySaver" or "Recycler" popular in the Los Angeles area), to see what the market price seems to be - and go lower for fast results.

Win, Win, Win!

Not long ago I sold my first computer, a ten-year-old TRS-80 Model I Level II 16K micro (originally $800) with a Stringy Floppy Drive (originally $250) and $400 worth of tape wafers, an 80-column 120 character-per-second electrostatic printer (originally $1200, but I got it used for $250), a special serial interface ($25), monitor, cables, and operating manuals. That adds up to over $2600 worth of original-cost items, and I sold it all for $300. The buyer wanted to learn Z80 microprocessor assembly language programming, and was delighted with his purchase.

The point is, I got $300 I didn't have before, it cleared the desk space for another computer I "had to have", and the buyer got what he wanted. A Win, Win, Win situation.

I recently sold a TRS-80 Model III, with CP/M added, for $500. This machine, with two 180K disk drives, originally sold for $2400, plus $800 for the CP/M. Yet the buyer was delighted! He had the need for both a TRS-80 Model III and CP/M - and he got them both in one machine! I bought this machine used about three years ago for $1000 - a good buy at the time. But times change, and the market value of older micros is continually dropping. For example, I've been offered a working TRS-80 Model II (originally sold for about $4000) for $200! This would be a great buy for someone with a working Model II that needs a backup. All the seller has to do is search out Model II owners.

So, bite your tongue, grin and bear it - and charge a price you KNOW is a good deal. If you are going to go to the effort to sell this thing, get it over with.

This does not mean you should put yourself in a bad negotiating position. I suggest you set the price about 20% or 25% higher than the absolute minimum you'll accept. This should be a fair, but not inflated price. You may find no reduction is necessary when you find an eager buyer.

Let The World Know

Next you need to get the word out that you have this micro for sale. On a national level, consider using the BCE service on compuServe, mentioned earlier.

For access to a more limited audience, advertise in a local newspaper. There are many papers that don't charge to place ads, but charge the buyer of the paper. They usually classify their ads in categories, making it easy for a prospective buyer to find exactly what he wants.

Use User Groups

You might also find a User Group that might be somewhere near you - one that supports the machine you want to sell. Local computer stores are one source for this information, and some microcomputer magazines have listings of User Groups. Contact a principal member of the group and ask them to announce the availability of your used machine at their next meeting, or to put it in their newsletter.

Some have small classified ads in their newsletters for this purpose. Also, you can attend a local meeting and make a "pitch" during the gathering. You might find, however, once you meet some local people that can help you, that you don't want to sell the machine after all!

Buyer's Experience

Although this is hard to avoid, try not to sell to someone who has not used this equipment before - especially one who has not used any microcomputer at all before. If you do, you must make it clear that you can't be involved in helping them learn how to use the machine. You could end up providing support for months!

I recently sold my Radio Shack MC-10 Microcolor Computer. It originally cost me $120, plus $50 for the extra 16K memory, plus probably $50 for extra books, programs and a vinyl carrying case. I sold the whole works, in mint condition, for $50. I made it clear I could not support the machine. The buyer has not called since. Now, this is a pretty simple machine, and I checked it out completely before selling it. I was willing to sell it to a novice since it was designed for beginners and had a good user manual.

The ideal buyer is one who already knows all about the machine you have and wants to add it to his own collection. Just a couple of weeks ago I sold a friend's TRS-80 Model I, with a 48K expansion interface, monitor and and five working disk drives (each in a separate case with power supply). Not only that, each of the card edges had gold-plated connectors added, and even a parallel printer cable was included. Total price: $300. The buyer, it turns out, really was only interested in the disk drives, and would have paid $300 for those alone. He ended up with the entire computer - as far as he was concerned - for nothing! I suppose if I were a sharp negotiator I could have started with $500 as the advertised price - but would the buyer have even called at that price? This way, my friend was happy to sell his computer, I was happy to help him, and the buyer was ecstatic. Win, Win, Win again.

Demonstration

Be prepared, if asked, to demonstrate the machine. I have a table set up in the garage with a six-outlet power extension, and it makes a great place to set-up and run a machine I want to sell. I personally will not sell a machine that is not working properly - demonstration or not - unless any defects are clearly described to the purchaser.

Buying A Used Micro

If you are a Never-Say-Die computerist and you want to add backup units or add-ons to your present machine, the process is similar. Check with the BCE on CompuServe. Advertise in a local paper under "Microcomputer Wanted". Join a User Group that supports the machine you want to buy. Many members may be thinking of "moving up" and would be willing to sell some or all of what they have to help pay for their next computer. Insist on a demonstration, with your own tapes or disks. If the distance is too great for a demonstration, specify a bill of sale guaranteeing refund if the unit is defective or not as specified.

And don't be afraid to ask for a price reduction. Most people overvalue when they sell and undervalue when they buy. There is usually a happy middle ground between buyer and seller.

Be careful of deals that sound too good to be true. Some equipment being offered for sale is stolen. If a computer, printer or monitor has a serial number removed, it might be stolen. If there is no documentation with the unit, this sometimes could mean its stolen, but it is not unusual for used equipment to have no paperwork. It is always best to ask for an original bill of sale when buying used equipment, but in the real world these have usually been lost.

Recently, since I'm the phone contact for a local TRS-80 user group, I got a call from a young man who had inherited a TRS-80 64K Model 4 with two disk drives. He had no manuals, documentation or programs, no micro experience, and no interest in keeping it. He was asking $150 for this machine that originally sold for $1800. I had him bring it over and checked it out. It had a serial number, and was like new! I didn't have the heart to ask him to drop the price. I gave him the $150 and we were both happy. It is now my most-used machine. Win, Win!

New User

If you're a brand new computer prospect and have never had a machine of your own, ask the help of a knowledgeable friend - after you have made it very clear to both yourself and your friend what your NEEDS are. Step One is determining your purpose, Step Two is finding the software that will satisfy Step One. Step Three is getting the hardware necessary to run the software.

Summary

Selling your used computer can bring in some extra money. Make sure everything works, set a low price, and get the word out. Be honest and you'll avoid a problem later on.

Buying a used computer can save you a bundle compared to a new machine. But you have to do a little homework. And it helps if you're a little paranoid, since some of what's out there for sale is defective or stolen.

• MODEL III

TRSDOS 1.3. CORNER

by Lance Wolstrup

From time to time TRSTimes will try to do something special for the ones of our readers who are still working with Model III, TRSDOS 1.3. We have gotten many letters urging more coverage of this DOS, and never let it be said that TRSTimes is not sensitive to reader requests.

TRSDOS 1.3.

This operating system IS the standard DOS for the Model III and, unless you got your machine from a fast talking guy wearing a big hat at the Greyhound Bus Station, all Model III owners should have a copy. It came with the machine.

Though it lacks some of the features of the 'fancy' Model III alternative DOSes, such as LDOS, DOSPLUS, NEWDOS/80 and MULTIDOS, it is still a GOOD operating system and, obviously, many folks are satisfied with it. It does, however, have some quirks that are irritating. This article will discuss AND fix some of these irritants.

BACKUP LIMITED DISKETTES

Many of us bought commercial software from Radio Shack, only to find out that we were only able to make a certain amount of 'backups' of these diskettes. It was Radio Shacks way of keeping us honest, to keep us from giving away copies to all our friends.

OK, for most of the reasons given in the various publications over the years, I agree that we shouldn't pirate software. However, I do strongly believe that when you own a legitimate copy of a program, you should be able to make as many copies as you like - for personal use.

When my LAST copy of ZORK, or other protected disk, blows up, the disk refusing to boot, I get upset. The only resource is to fork over more cash to Radio Shack for a fresh copy, and in this stage of the Model III's life, it probably isn't even available. So, before this happens to you, let's fix this problem once and for all.

UNLIMITED BACKUPS

The boot sector (track 0, sector 1) on all TRSDOS 1.3. system diskettes has a byte dedicated to telling the BACKUP utility how many backups are allowed. This byte is located at 22H (34 decimal). If it contains FFH, BACKUP will allow unlimited backups. Any other number means that you have that amount of backups left. Upon completion of a backup, this number (if not FFH) will be decremented by 1 and written back to the boot track. When it reaches 0, TRSDOS will go through the motions of a backup, but the 'backup limited' files will not be copied.

If you own, and are familiar with, a 'zap program', simply replace whatever number is at 22H on track 0 - sector 1 with FFH and write it back to the disk.

Realizing that most readers will not own, nor be familiar with, a 'zapper', we will take care of the fix in a different manner. Instead of rewriting the boot sector we will fix the BACKUP utility so that it doesn't care how many backups the boot sector says you have left.

Here you have to be extremely careful. Use a backup of the original disk, if possible, or if you have any backups left from the original, make one now. If all you have is the original with no backups left, use it - at your own risk!!!

Insert the disk in drive :0 and reboot. When the 'TRSDOS Ready' prompt appears, type the following:

PATCH *7 (ADD = 528E,FIND = C8,CHG = C9)

When you are ABSOLUTELY SURE that you have typed the above correctly, press the (ENTER) key. If you are successful, you should see the message 'Patch Made' displayed on the screen, followed by TRSDOS Ready. (If a mistake in the above patch was made, TRSDOS will display 'String NOT Found').

Assuming that the patch was made correctly, you are now able to make as many backups as your heart desires. Now, that wasn't difficult, was it!

For the ones of you who would like to know what took place, here is a short explanation:

The PATCH command allows us to change the contents of a disk file. You must specify the name of the file, the address of the data you wish to change, the current data at that address, and the data you wish to replace it with.

Let's take this information and apply it to what we just did:

PATCH - this is the command.

*7 - this is the filename. The BACKUP utility is found in overlay 7. The ability of PATCH to modify overlays is not documented in the manual. It works just fine, though. I guess they just didn't want us to know.

(ADD = 528E, - this is the address in overlay 7 where BACKUP has just finished checking to see if you have more backups left. It makes a comparison.

FIND = C8, - this is machine code for the assembly language instruction RET Z. This means that if the comparison of 'are infinite number of backups allowed' is TRUE (zero), we skip the routine that goes on to limit the backups. If the comparison is FALSE (non-zero) we don't skip and instead go on to that 'nasty' routine.

CHG = C9) - this is the fix. C9 is machine code for the assembly language instruction RET. The difference between RET Z and plain RET is that RET Z is conditional. That is, it acts one way if the comparison is TRUE, another way if the comparison is FALSE. RET is unconditional. It doesn't care what the result of the comparison is. It skips the routine regardless. Success!

MORE PATCHES
Getting rid of useless prompts.

Another feature that has annoyed me for some time is, when formatting a diskette, you are prompted for the disk name. Now, I don't know if you like this feature and faithfully name each disk for some unknown, sinister purpose. I don't like it. It forces you to type something and then pressing the (ENTER) key. Simply pressing (ENTER) for a default name will not work; it redisplays the 'Diskette Name?' prompt.

Here is the patch that will completely bypass the 'Diskette Name?' prompt. Not only will you not be prompted ever again, FORMAT will use 'TRSDOS' as a default disk name.

PATCH *7 (ADD = 5666,FIND = 21,CHG = C9)

As in the explanation of the previous patch, PATCH is the command, *7 is the filename of overlay 7, which contains the code to both the BACKUP and FORMAT utilities.

(ADD = 5666, this is the beginning of the routine to prompt for the disk name.

FIND = 21, - 21 is machine code for the assembly language instruction for LoaD register HL with the number in the next byte.

CHG = C9) - just as in the previous example, C9 is machine code for an unconditional RETurn. We don't need any of the information in the routine and since it was CALLed by the main program, we can simply RETurn immediately thereby bypassing all the code that produces the disk name prompt and checks to see if one has been typed. (CALL is the assembly language instruction that works just like GOSUB in Basic.)

Next we will completely disable all passwords. This means that you will have complete access to all files, irregardless of whether or not they are password protected.

The patch is:

PATCH *2 (ADD = 4ED4,FIND = 20,CHG = 18)

Explanation: PATCH - the command

*2 - file name of overlay 2, which contains the "OPEN" and "INIT" routines.

(ADD = 4ED4, - In order to do almost anything with a file, it must first be opened. DOS, of course, handles this for us. At address 4ED4H in overlay 2, DOS is on its way to open whatever file was specified from the DOS command line or by some application program. Before the file can be opened, TRSDOS checks memory location 42FFH to see if password check should be skipped.

FIND = 20, - Address 4ED4H makes a comparison on the value found in bit 0 of 42FFH. In essence DOS is asking 'shall I skip the password check?'. If the bit 0 is OFF (0) the password chek is performed. If bit 0 is ON (1) the check is skipped by jumping over the code. 20 is machine code for the assembly language instruction 'JR Z,offset found at next byte'. This is pretty much like when you say in Basic: IF X = 1 THEN 200. These instructions are conditional jumps. They will branch to one section of the program if the value is 0, branch to another if the value is 1.

The easiest way to handle this is to make the conditional branch unconditional. In other words, we are telling DOS that we really don't care if the bit 0 of 42FFH is ON or OFF. Either way skip the password check. We do this by changing 20 to 18.

CHG = 18) - this is machine code for the assembly language instruction 'JR offset found at next byte'. The JR instruction is unconditional. DOS has no choice but to jump over the password check code.

We have effectively disabled all file password checking. Now we will disable the password check in the BACKUP utility.

PATCH *7 (ADD = 55A8,FIND = 28,CHG = 18)

Explanation: By now, I am sure, we understand that PATCH is the command and *7 is the file name for overlay 7.

(ADD = 55A8, - here BACKUP makes a comparison if the disk password and the password you typed in matches.

FIND = 28, - 28 is machine code for the assembly language instruction 'JR Z,offset found at next byte'. If the answer to the question 'do the passwords match?' is TRUE (0), we then jump over the section of code that handles non-matching passwords, and go directly to the actual backup procedure. If the answer is FALSE (not 0), we do not jump. Instead we go, by default, to the code for non-matching passwords that gives us an error message and, eventually, takes us back to DOS without performing the backup. Again 'JR Z,offset found at next byte' is a conditional instruction.

CHG = 18) - by changing the conditional 'JR Z,offset found at next byte' to an unconditional 'JR offset found at next byte', we tell BACKUP not to care if the passwords are not identical. Perform the backup anyway, Charlie!

You can now type any password of 8 characters or less, or simply press (ENTER), to the prompt. Backup will be performed as if the correct password had been typed. However, there doesn't seem to be much sense in prompting for a password that doesn't matter, so let's finish up this installment by disabling the password prompt from both the BACK UP and FORMAT utilities.

PATCH *7 (ADD = 565A,FIND = 06,CHG = C9)

Again the file name indicates overlay 7.

(ADD = 565A, · here we are at the first common code in the subroutine that handles the password prompt for both BACKUP and FORMAT.

FIND = 06, · 06 is machine code for the assembly language instruction LoaD register B with the value of the next byte. Since the entire routine doesn't do anything but issue the prompt, we can skip it entirely.

CHG = C9) · As mentioned earlier, C9 is the RETurn instruction which sends the program flow back to the main body of the program. (like a RETURN from GOSUB in Basic.) By returning, we skip the prompt. When you now use the FORMAT utility, one of two things will occur: 1. If you type FORMAT :1 (ENTER) and the disk in drive :1 is unformatted, the format will take place without any further action by you.

2. If you type the above command and the disk in drive :1 has data, you will be told and prompted if you want to format it. Answering 'Y' will cause the format to occur without any further action by you.

The BACKUP utility now works as follows:

1. If you type BACKUP :0 :1 (ENTER) and the disk in drive :1 is unformatted, the disk in drive :1 will be formatted and then the backup will take place. No action on your part is neccessary.

2. If you type the above command and the disk in drive :1 is already formatted, you will be prompted with: 'Do you wish to RE-FORMAT?'. Typing 'Y' will format the disk again, then perform the backup. Typing 'N' performs the backup. Either way, no further action is neccessary.

In closing, let me give a list of Radio Shack's lineup of 'backup-limited' software:

ZORK1 · DEADLINE · WITNESS · STARCROSS · SUSPENDED · PLANETFALL · ECHANTER · MONTY PLAYS MONOPOLY · MONTY PLAYS SCRABBLE · PFS file, PFS report, XENOS and probably many more whose titles I have forgotten.

Also, the power you have been given here is NOT intended to licence anyone to pirate software or invade the privacy of other peoples files. It is intended solely to help you manage your own legitimate software and private files. Enough said.

As time permits, I will play around some more with TRSDOS 1.3. Maybe some other interesting things will appear in future issues.

TRS-80 Software from Hypersoft.

Read CP/M CoCo & PC disks on your TRS80

Use HYPERCROSS to COPY files between TRS-80 disks and those from many CP/M and IBM-PC type computers on your TRS-80 I, III, 4/4P or Max-80. You can FORMAT alien disks, read their directories, copy files to and from them, copy directly from one alien disk to another. Converts tokenized TRS80 BASIC to MSDOS or CP/M as it copies.
Formats supported: IBM-PC and MS-DOS including DOS 1.1, 2.0-3.2 Tandy 2000, single and double sided, 3.5 and 5 inch. CP/M from Aardvark to Zorba. CoCo format on XT+ version.
HyperCross 3.0 PC reads popular MSDOS 1.1-3.2 formats **Order SX3PCM1, SX3PCM3 or SX3PCM4** $49.95
HyperCross XT/3.0 reads 90 different CP/M and PC formats **Order SX3XTM1, SX3XTM3 or SX3XTM4** $89.95
HyperCross XT/3.0-Plus. Reads over 220 formats inc CoCo **Order SX3XTM1+, SX3XTM3+ or SX3XTM4+** $129.95
Specify TRS-80 Model I (needs doubler), III, 4/4P or MAX-80. Dual model versions e.g. Mod 3/4 on one disk add $10 extra.

Amazing HYPERZAP 3.2G Disk Magic!

Do you want to backup, fix or modify a disk - if so then **you need HYPERZAP!**. More than just another disk copying program - it is the program for analyzing, copying, repairing, creating floppy disks of all kinds. It works with TRS-80 formats as well as many others such as CP/M, PC, CoCo etc.Designed to handle mixed density sectors on any track in any sequence. Many features for reading, writing, editing track and sector data. **Make your own self booting disks. Autopilot mode** learns, saves and repeats procedures. Disk comes with fascinating examples. Use Hyperzap as a learning tool, find how things are done!.
HYPERZAP 3.2G - nothing else even comes close! Order # HZ32 - one version runs on all Model I/III/4/4Ps $49.95

Other TRS-80 Programs

FORTH: Mod 1/3 $49.95, Enhanced Model 4 version: $59.95
LAZYWRITER Word Processor for Model 1, 3 or 4 .....$109.95
MultiDOS 2.1 1988 and beyond ! Model 1 or 3 .....$79.00
MultiDOS 2.1 64/80 version for Model 4 .....$89.00
Mysterious Adventures - Set of 10 for M1, 3 or 4(3) complete $49.95
NUTRITION Analyze your diet, with database, Model 4 only $49.00
TASMON debug trace disassemble TASM1 TASM3 or TASM4 $49.95
TMDD Memory Disk Drive for NewDOS 80/Model 4 users $39.95
XAS68K 68000 Cross Assembler, specify Mod 1, 3 or 4 .....$49.95
ZEUS Editor/Assembler specify Model 1, 3 or 4 ..... $74.00
ZIPLOAD fast load ROM image, DOS & RAMDISK on your 4P $29.95

Run Model 4 Software on a PC with PC-Four !

Now you can run your favorite TRS-80 Model 4 programs on a PC !. **PC-Four** is a program that makes your PC or Compatible behave like a 128K TRS-80 Model 4 complete with operating system, Z80 microprocessor that can run many true Model 4 programs such as **ALDS, ALLWRITE, BASCOM, BASIC, C, COBOL, EDAS, ELECTRIC WEBSTER, FED, FORTRAN, HARTForth, Little Brother, MULTI-BASIC, MZAL, PFS FILE, PASCAL, Payroll, PowerMail, PROFILE, SUPERSCRIPSIT, TASMON, VISICALC, ZEUS** and more.

Runs on PCs, PS/2s, compatibles and laptops with at least 384K of memory. ONLY emulates Model 4 mode of Model 4. To use it you must transfer your old files to MSDOS disks using **PCXZ or Hypercross.**

Prices: Order #PC4 $79.95 alone, #PC4H $104.95 with Hypercross SX3PCM4, #PC4Z $119.95 with PCXZ. Available on 3.5" disk format.

PCXZ reads TRS80 disks on a PC

PC Cross-Zap (PCXZ) is a utility that lets you **copy files to or from** TRS-80 disks on a PC or AT. Transfers BASIC, ASCII and Binary files. **Converts BASIC and text files automatically.** You can also format a disk, copy disks, explore, read and write sector data, repair bad directories and much more. **Supports:** all double density Model I, III and 4 formats. **Requires:** PC, XT, AT or compatible. **You must have at least one 5-1/4" regular or high density drive and 256K memory. Not for PS/2s: Order # PCXZ** **$79.95**

Hypersoft

POB 51155, Raleigh, NC 27609

Orders: 919 847-4779 8am-6pm, Support 919 846-1637 6pm-11pm EST

MasterCard, VISA, COD, Checks, POs. Add $3 Shipping, $5 2nd day

from HOLLAND

• **Model I/III · Newdos/80**

MACROKEY

by Ruud Broers

Just what is a macro?

The Illustrated Dictionary of Microcomputers (2nd edition) gives the following definition:

Macro A group of often-used instructions treated as a unit entity.

In other words, a macro is a form of shorthand. A long string of commands can be programmed into a macro, which can then be accessed and executed by hitting a special sequence of key strokes.

This is exactly what MACROKEY is all about. It is an assembly language utility for Model I and III running NEWDOS/80 that allows you to execute a long series of commands with just two keystrokes.

Here is what to do:

First type in the program listing using an Editor/Assembler; then assemble it as MACRO/CMD.

When ready to use the program, type MACRO <ENTER>. You will immediately be returned to the NEWDOS/80 READY prompt.

At this point activate the program by pressing the 'macro control key'. On a Model 4 running in Model III mode this control key is: <F1>.

On a straight Model III and the Model I the control key is: <SHIFT> <DOWN ARROW> <Q>.

A large white character - CHR$(143) - will be displayed, indicating that a macro may now be defined.

Simply type the word, command, or string of commands you wish to define as a macro.

When done press the control key once again. The large white character reappears indicating the end of the macro. Now press <ENTER> and you will be back at the NEWDOS/80 READY prompt. Your macro has been defined.

Examples of macro definitions:

<control key> DIR,1,S,I,A <ENTER> <control key> <ENTER>

This programs the macro to display the directory of all files on drive :1 whenever the control key is pressed twice. Notice that <ENTER> is a legitimate command inside the macro.

<control key> COPY,0,1,,NFMT, CBF, CFWO, SPDN=4 <ENTER> <control key>

This will copy individual authorized files from drive :0 to drive :1 without first formatting the destination disk, simply by tapping the control key twice.

<control key> DATA <control key>

If you are typing in a Basic program with a lot of DATA statements it would save a lot of time by defining the macro as <space> DATA <space>.

Now, type the linenumber, press the control key twice and the word DATA is there, one space away from the linenumber and with a trailing space, ready for you to type the actual data information.

The possibilities are almost endless, but the real beauty of the program is that you can redefine the macro on the fly just by pressing the control key once, typing the new macro, again pressing the control key once, and then hitting the <ENTER> key.

The macro will remain in memory until you redefine it or reboot. Try MACROKEY, you'll like it.

```
MACRO
00100 ;
00110 ; MACRO 1.0 for Model I/III/(4)
00120 ; update 11/17/87
00130 ; for use with  NEWDOS 80
00140 ;
00150 ; Ruud Broers, Penningburg 73
00160 ; 3437SJ Nieuwegein, Holland
00170 ;
00180 CALL    EQU   0CDH
00190 DOS     EQU   402DH     ;return address
00200 KBDCB   EQU   4016H     ;keyboard DCB
00210CHROUT   EQU   33H       ;display chr in a
00220  F1     EQU   60H       ;for model 4(p)
00230 ;                       ;(F1)
00240 ;F1     EQU   11H       ;for model I/III
00250 ;                       ;(ctrl Q)
00260 BS      EQU   08        ;back space
00270 BEEP    EQU   07        ;beep
00280 BLOK    EQU   8FH       ;block-character
00290 ;
00300 ;
00310         ORG   0FF01H
00320 INIT    LD    HL,(KBDCB) ;activate new
00330         LD    (PATCH),HL ;keyboard drvr
00340         LD    (PATCH2),HL
00350         LD    HL,DRIV
00360         LD    (KBDCB),HL
00370         JP    DOS       ;ready
00380 ;
00390 ;
00400 DRIV    LD    A,(ROA)   ;read-out active
00410         OR    A         ;?
00420         JR    Z,NRO     ;if no, jp
00430         LD    HL,(POINTR) ;resume
00440         LD    A,(HL)
00450         OR    A         ;end?
00460         JR    Z,LRO     ;if yes, then flag
00470         INC   HL        ;if no, point to
00480         LD    (POINTR),HL ;next chr
00490 LRO     LD    (ROA),A
00500         RET
00510 ;
```

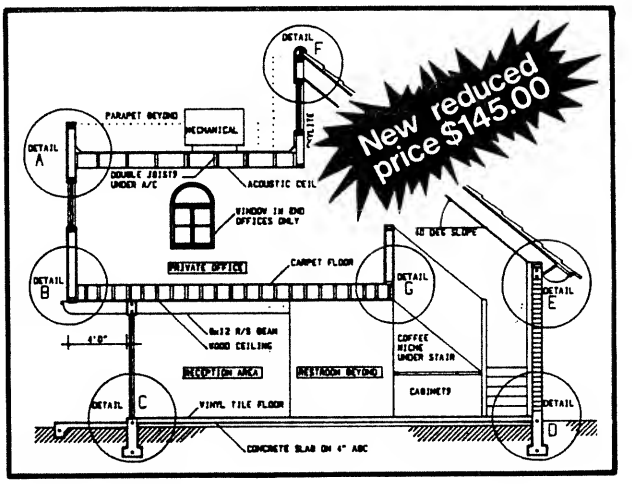
```
00520 ;
00530 NRO       DEFB  CALL      ;fetch key
00540 PATCH     DEFW  0000
00550           CP    F1        ;used f1?
00560           RET   NZ        ;if no, than ready
00570           LD    C,A       ; = f1 ; to test for 2x f1
00580           LD    A,BLOK
00590           CALL  CHROUT    ;print character
00600           LD    HL,MACRO  ;point to macro
00610           LD    (POINTR),HL
00620 PBC       PUSH  BC        ;store bc
00630           PUSH  HL        ;store hl
00640 WKEY      DEFB  CALL      ;call keY
00650 PATCH2    DEFW  0000
00660           OR    A         ;key pressed?
00670           JR    Z,WKEY    ;if no, than again
00680           POP   HL        ;restore hl
00690           POP   BC        ;restore bc
00700           CP    C         ;was it f1 ?
00710           JR    Z,SROO    ;if yes, then read-out
00720           CALL  CHROUT    ;else define macro
00730           LD    C,0
00740           CP    F1        ;finished defining?
00750           JR    Z,MDD     ;if yes, jp
00760           LD    (HL),A    ;store character
00770           INC   HL        ;next position
00780           LD    A,(HL)    ;test end buffer
00790           CP    0FFH
00800           JR    Z,MDD0    ;if yes, jp
00810           JR    PBC       ;fetch next chr
00820           MDD0  LD A,BEEP ;buffer full
00830           CALL  CHROUT    ;beep
00840           DEC   HL
00850 MDD       LD    (HL),0    ;put end marker
00860           LD    HL,MACRO
00870           LD    (POINTR),HL ;set pointer to begin
00880           LD    A,BS      ;only for model 4(p)
00890           CALL  CHROUT    ;only for model 4(p)
00900           LD    A,BLOK    ;end defining mode
00910           CALL  CHROUT
00920 EXIT      XOR   A
00930           RET
00940 ;
00950 ;
00960 SROO      LD    (ROA),A   ;set read-out flag
00970           LD    A,BS      ;dispose of last chr
00980           CALL  CHROUT
00990           JR    EXIT
01000 ;
01010 ;
01020 ROA       DEFB  00        ;read-out 0 = active flag
01030 POINTR    DEFW  MACRO
01040 MACRO     DEFM  'Not yet defined a macro .'
01050           DEFB  BEEP
01060           DEFB  00        ;end defining
01070           DEFM  '                    ';3 times
01080           DEFM  '                    ';32
01090           DEFM  '                    ';chrs
01100 ZZ        DEFB  0FFH      ;end buffer
01110           END   INIT
```


Professional results with TRS-80

Fully supported products for

Models III and 4 4p 4D

PC/XT/AT, Tandy, and 100% compatibles (MSDOS)



xT.CAD PROFESSIONAL by Microdex. Computer Aided Drafting software for technical production and education. Create, edit, modify precise drawings, details. Features include overlays, grids, cursor snap, zoom, pan, block copy, enlarge, reduce, rotate, mirror, clip, merge, text labels, more. Requires hi-res screen and RS-232 interface. Output to pen plotters. Input from keyboard, or optional digitizer or mouse. Friendly, competent support since 1984. Software is backup-free. Was $345 in 1986 catalog.
Model III or **4 4p 4d** or **MSDOS** $145.00

xT.CAD BILL of Materials by Microdex. Software utilizes text labels from xT.CAD drawings to automatically generate invoices, parts requests, shipping lists, etc. Includes a mini-editor for customizing line printer output.
Model 4 4p 4d or **MSDOS** $ 45.00

CASH PROFESSIONAL by Microdex. Bookkeeping software with automatic double-entry ledger distribution in user-definable accounts. Reports by period, account, project, etc. Ideal for small business, professional or personal accounts.
Model 4 4p 4d or **MSDOS** $45.00

S/XT software by Microdex. Enables disk directory review and special character printing from within standard Scripsit.
Model III or **4 4p 4d** $15.00

GRAFYX Solution by Micro-Labs. Easy to install board provides hi-res similar to Radio Shack boards. Includes popular GBASIC software and manual. Supports xT.CAD and other graphic programs.
Model 4 4p 4d (640x240 pixels) $125.00 ~~$145.00~~

MOUSE interface by Micro-Labs connects to 50-pin I/O port and allows the use of Tandy Color Mouse 26-3025 or 26-3125 (not included) with xT.CAD and other programs.
Model III 4 4p 4d $115.00

Microdex Corporation
1212 N. Sawtelle
Tucson AZ 85716

602/326-3502

Write or call for details.

MICRODEX

from ENGLAND

- **Model 4 - TRSDOS 6.2 & LS-DOS 6.3.**

USING SYS13/SYS

by Gordon Collins

To save anyone doing a lot of reading , and then finding that they are not able to use this, can I now point out that the following will only work with TRSDOS 6:2:xx and LS-DOS 6:3:xx. Also I shall be keeping to a very narrow path whilst writing, I would get lost myself if I did not. If anyone understands more of this than me, please feel free to take detours. I shall go through this in stages so that anyone wanting to branch off can do so, you may find out more than I have. I am not a DOS expert, but I am willing to learn whilst listening to others.

My natural inquisitive instinct was aroused, when I read, against a reference to something called an "EFLAG$", "that this byte is used to indicate the presence of an Extended Command Interpreter (ECI) program in the SYS13/SYS slot". Turning to the Model IV Owner's Manual revealed nothing. (Because it was written before TRSDOS 6:2?). The Technical Manual for TRSDOS 6:2, page 262, has 5 lines which meant nothing to me, and page 190 tells you to remove the file if you have not implemented an ECI. Have I missed something somewhere?

Over a period of time I found out the following, which is a very brief synopsis of all my workings in the land of the unknown, I wanted to copy VC/CMD to SYS13/SYS and have it work - the same basic procedure can be used for other /CMD files. If you have disk drives of other capacities than those indicated, and are sure enough in yourself to modify my path, then please feel free.

BACKUP your system disk to include an original copy of SYS13/SYS. This backed up system disk should now be put in drive :0. REBOOT and then type:

DIR SYS13/SYS:0 (S) <ENTER>

You should see that the "File Size" is 1.5K.

Make at least 27K of free space on the disk (Mod. 4 VC/CMD is 27K in size).

Type: **PURGE :0** <ENTER> and answer **Y** to a number of files.

<BREAK> when you have removed enough files (I answered **Y** to: BREF/CMD, DOS/HLP, HELP/CMD and TAPE100/CMD).

Then type: **FREE :0** <ENTER> as a check.

Using Model 4, VC/CMD on a disk in drive :1 (I am using this file as an example only, as almost any /CMD file can be copied to SYS13/SYS) type in the following:

COPY VC/CMD:1 TO SYS13/SYS.LSIDOS:0 (C=N) <ENTER>

As a check, type:

DIR SYS13/SYS :0 (S) <ENTER>

You should now see "File Size" as 27.00K. Still with me?

Type: **DIR :0 (S,I)** <ENTER>

You should not have found VC/CMD on this disk.

NOW at DOS READY, type <*> <ENTER>, yes just an asterisk and <ENTER>, VisiCalc should arrive on your screen.

You have just executed an ECI Program!!

This is where it got a bit more difficult for me. The above is not going to move mountains, but please keep reading. Whilst in VisiCalc ENTER,

</>, <S>, <Q>, <Y>.

You should now be at DOS Ready, you could have just rebooted.

At DOS Ready type: **MEMORY (ADD="E")** <ENTER>

A line on the screen should read:

X'006E' = 110 (X'0000', ... etc.

What we are interested in are the first two X'nnnn's, this shows that the EFLAG$ is not set. The EFLAG$ being at MEMORY ADDRESS X'006E' and is X'00'.

Again at DOS Ready type:

PATCH SYS0/SYS.LSIDOS:0 (D00,7A=FF: F00,7A=00) <ENTER>

This has now permanently zapped the EFLAG$ byte in the SYS0/SYS File and since SYS0/SYS is always in memory from bootup, you will now have to reboot to effect this zap, so now please reBOOT. If you did that, VisiCalc should have appeared on your screen. This is another way of executing an ECI. Whilst SYS0/SYS remains zapped on this disk, on each BOOT'ing, DOS will want to load SYS13/SYS, you will not see "DOS Ready" again.

Whilst VisiCalc is on the screen, please type the following: </>, <S>, <E>, and in the edit line type, **MEMORY (ADD="E")** <ENTER>, you will now see that the byte at memory address X'006E' has changed to X'FF'. Hit any key to reformat the screen.

We now have one big problem, try exiting VisiCalc by typing, </>, <S>, <Q>, <Y>, as we did above. So something just wants to keep loading the ECI, its that EFLAG$ byte we zapped, so with VisiCalc now on screen type:

</>, <S>, <E>

In the edit line type:
MEMORY (ADD = X'6E',B = X'00')

You will see that we have rezapped MEMORY ADDRESS X'6E' to X'00'. Hit any key to reformat the screen and exit VisiCalc as above, and you should get to DOS Ready.

The next stage is so that we do not remain in that endless loop on trying to exit SYS13/SYS. This zap is for VC/CMD, record X'0068' and starting at byte X'00' you should find:

02 02 00 30 6D B6 6D B6 6D

change these nine bytes to:

01 03 6E 00 00 02 02 00 30

This zap will have SYS13/SYS reset address X'6E' to X'00' in memory when loading.

A better solution is to apply the following patch after any /CMD has been copied to SYS13/SYS. This will take a few moments and will add a few bytes onto the end of SYS13/SYS. Now when SYS13/SYS is loaded MEMORY ADDRESS X'6E' will be reset to X'00' so that an exit can be made to DOS Ready from the SYS13/SYS file.

PATCH SYS13/SYS (X'006E' = 00)

Just a further comment before ending, at DOS Ready type: **AUTO DDUTY/CMD**

On BOOT'ing up DDUTY/CMD is loaded, also SYS13/SYS, the second partition of DDUTY is then available for something else.

N.B. LeScript, when in SYS13/SYS has to be zapped at record X'12'. **Find "LESCRIPT/CMD" and change to SYS13/SYS**, as LeScript looks for its own name before loading, or you could have the file "LESCRIPT/CMD" also on the disk - but this wastes space.

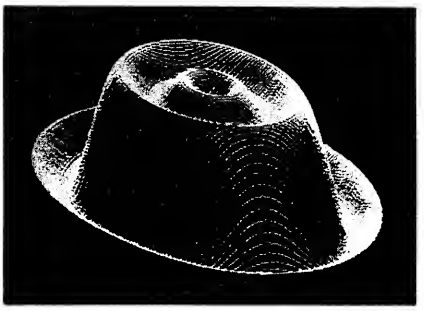
Another N.B., is that when using LeScript with DDUTY, note the comment on Page 97 of the LeScript manual.

The various manual references made above may not be applicable to all world wide, as there does seem to be various local differences.

Gordon Collins is the Editor of NATGUG News.
National Tandy & General User Group
11 Elizabeth Road.
Sutton Coldfield
West Midlands. B74 5AR
England

Grafyx Solution™ Save $170.00

Hi-Resolution Graphics for Mod 4/4D/4P/III



Superior Hardware. The Grafyx Solution provides 153,600 pixel elements which are arranged in a 640 × 240 or on the Model III a 512 × 192 matrix. Hundreds of new business, personal, engineering, and educational applications are now possible. The hi-res display can be shown on top of the standard display containing text, special characters, and block graphics. This simplifies program debugging, text labeling, and upgrading current programs to use graphics. The Grafyx Solution fits completely within any tape or disk based Model 4, 4D, 4P, or III. Installation is easy with the plug-in, clip-on Grafyx Solution board.

Superior Basic. Over 20 commands are added to the Basic language. These commands will set, clear or complement points, lines, boxes, circles, ellipses, or arcs. The hi-res screen can be printed on any of 30 popular printers or saved or loaded to disk without leaving Basic. Areas may be filled in with any of 256 patterns. Sections of the screen may be saved and then put back using any of five logical functions. Labels can be printed in any direction. The viewing area can be changed. The entire screen can be complemented or cleared. Graphics Basic provides dot densities of 640 × 240, 320 × 240, 160 × 240, and 160 × 120, all of which can be used in the same display.

Superior Software. The board comes with over 40 programs and files which make it easier to use, serve as practical applications, demonstrate its capabilities, and serve as programming examples. The software works with TRSDOS 1.3, 6.1.2, 6.2, 6.3; Dosplus 3.4, 3.5, 4; LDOS; and Newdos80. The Grafyx Solution is also supported by 30 optional applications programs: Draw, Bizgraph, xT.CAD, 3D-Plot, Slideshow, Mathplot, Surface Plot, Chess, etc.

The Grafyx Solution package is shipped complete for $129.95 (reduced from $299.95). The manual only is $10. Payment may be by check, Visa/MC, or COD. Domestic shipping is free on pre-paid orders. Texas residents add 7% tax.

MICRO-LABS, INC. 214-235-0915
902 Pinecrest, Richardson, Texas 75080

Tim's PD Express

High-Rez 80

by Timothy Sewell

This month I am going to cover the world of HIGH RESOLUTION software that is available in the Public Domain. Please note that in order to use this software, you MUST have an optional High Resolution board installed in your Model 3 or Model 4. The boards are available from both Radio Shack and Micro-Labs, and they add a whole new dimension to your computer.

For those of you who like to do drawing, there are a few choices available to you...

TRSDRAW

Paul Bradshaw has written what is considered to be the definitive Public Domain drawing program. TRSDRAW is a powerful full-screen graphics editing program for use with the Radio Shack High Resolution graphics board. The program will not work with graphic board from Micro-Labs.

Some of the more impressive features included in TRSDRAW are the ability to remember up to 26 different "points" of reference for drawing, and an on screen Help function that "windows" a help screen over the graphic without disturbing what you're working on. Other features include drawing boxes, drawing circles, storing small "blocks" of graphics into memory for manipulation, and screen inversion.

What is also nice about TRSDRAW is the ability to recover a previously stored graphic if you make a mistake. This "Whoops" function lets you recall from 2 banks of stored memory so if you screw something up, it can be returned to the state it was last saved in.

The one 'draw' back to TRSDRAW is the fact that you can't print a screen directly from the program. Paul has promised to include this function in future releases but so far support for this wonderful program has not impressed the author enough to warrant upgrading. Maybe if more users support his efforts he might change his mind.

DRAW

Though not as powerful as TRSDRAW, DRAW by Scott McBurney has the unique ability to translate DOTWRITER graphic fonts on to the High Resolution screen. The actual drawing functions are minimal and the author even suggests that you use one of the other available drawing programs to put together your graphics and then use DRAW to add your letter fonts. DRAW is written in Micro-Lab's GBASIC but appears to work on both graphic boards.

TRSTEXT

George Madison decided that he enjoyed the advanced functions of TRSDRAW and also enjoyed the ability to use DOTWRITER fonts on the screen that DRAW provides. Problem was that he hated having to switch between the two programs. George's answer is TRSTEXT. TRSTEXT is NOT a drawing program. It is a program that will MERGE new code into TRSDRAW to give it the DOTWRITER capabilities of DRAW. You now have the best of two worlds in one program (Please note that there are several DOTWRITER format graphic fonts available in the Public Domain).

If viewing and printing pictures are what you enjoy, there's enough programs out there to keep you busy for a long time (I can attest to this!).

RLE stands for Run Encoded Length. It is a method of viewing and transfering graphics that was popular on Compuserve. The format was created so that weather maps could be viewed on computers with the ability to view High Resolution Graphics. Soon the format expanded beyond maps and just about any subject was game for this format. Since there are no Communication programs for the Model 3 or 4 that incorporate High Resolution Graphics, seperate programs for viewing and printing these files were created.

GUTIL

One of the first full function programs to handle RLE format files is GUTIL/CMD. GUTIL can load, save, encode, and decode RLE format files as well as HR format files (Files created with TRSDRAW, DRAW, PRODRAW, etc.). The programs ability to expand and move the file around faster then any program I've seen makes this a "must have" addition to anybody's library. I do not know who the author of this fine program is but I would like to.

RLE2BIN

Scott Swaine has written what I consider to be the best RLE file viewing/printing utility around. The program can handle RLE files in both BINARY and normal RLE format files (RLE files for the CoCo and Atari are in Binary format though they need a bit of correction before viewing). Scott's program also lets you dump the graphic on to your printer. They come out just fine on my DMP200. Two different versions

of the program are available one for the Model 3 and one for the Model 4.

RLEGEN

This program written by an unknown author allows you to save a file in RLE format on to disk. The program takes whatever High Resolution that is in memory and scans every 2nd column of 512 with 64 column margins. Not all pictures translate into RLE but it's fun seeing what effects can be created. This program is also available for the Model 3, as well as the Model 4.

A format of picture viewing that has taken the Model 4 world (if not the entire computer industry) by storm is the ability to view and print Macintosh created MacPaint format files. These READMAC files are popping up faster then I can download them and believe me when I say I have collected over 2000 different files. Please note that these programs are available for the Model 4 only. The model 3 does not have the graphic capability to handle MacPaint format files.

MACDISP

Scott Swaine ushered in the era of MacPaint file viewing with this little beauty. MACDISP will load a MacPaint file into memory and allow you to view the picture by scrolling it with the arrow keys (for a quick view from top to bottom you can use the CTRL and arrow keys together). Another great feature of MACDISP is when you want to load a file, you are promped for the drive the file can be found on. All files with a /MAC extention are displayed and you move the cursor to the desired file to be loaded, hit ENTER and the program works it's magic. MACDISP will also let you dump the picture to your dot matrix printer in 80 or 160 column mode.

HIRES

HIRES (not to be confused with the Root Beer of the same name) is written by Mel Patrick and showed up not to long after MACDISP. HIRES is a bit more elaborate though. It can also handle HR format files and also allows you to store the HR files into a compressed format for better disk storage. Viewing a MacPaint on HIRES is a bit different though, the program displays the picture in 3 seperate screens that are toggled rather than scrolled. Personally, I prefer to scroll/view my pictures, but that should not discourage you from seeking out this fine program. HIRES also give you a choice of printers to dump the picture to and will also dump the picture in reverse mode if desired.

For those of you without High Resolution boards who would like to see some of these Macpaint pictures, don't fret! Mel has written several different programs for the Model 3 and 4 that will print the file to your dot matrix printer without needing to have a High Resolution Board installed. Different programs are available for different printers.

So far I have concentrated on just picture drawing and viewing programs. Many other types of software are available for use with the High Resolution Board.

If you enjoy games and simulations, you will find the following programs of interest...

TRSSTORM is a Hurricane tracking program that is written by Dave Phillips. TRSSTORM allows you to track storms and recall it's travel path. Previously stored storm data files can be recalled and viewed for reference purposes. The program can track up to 59 different cities in the Gulf Coast area and you can manually add your own city if desired. TRSSTORM is available in both GBASIC and BASICG versions.

SHUTTLE is a BASICG program that draws a very detailed rendition of the space shuttle on your screen and then proceedes to go through the launch procedured step by step.

SATSONG, written by Robert Siegel and Bob Griggs is a very relaxing program to view. A space scene is generated on your screen and you see a planet with several moons in orbit. A rouge space ship sails the space ways and occasionally crashes into one of the moons. Very nice graphics and a joy to run.

CONCEN by Kerry Miles is a variation of the popular "Concentration" game. Instead of prizes like the TV game show, the "clues" are given as graphic patterns and you must search for it's match.

STOCKS is a High Resolution version of the Del Mar Stock Exchange simulation game is written by Alfred Funk (where is he now?). You can buy, sell, trade, and deal with brokers in this realistic game.

LUNAR LANDER by Ken Conaster is written for GBASIC and the Micro-Labs board. The game has five levels of difficulty and GLOADS the playing screen to give this game some of the finer graphics that can be found.

A new program that was uploaded to GEnie just a couple of days ago is a very well done Crossword puzzle game. **CROSSWRD** by Leonard Lorden will draw a pre-programmed puzzle on to your screen and allows you to toggle between the clues and the puzzle as you work it. A very impressive program worth searching for. If you donate $5.00 to the author and send him a disk, he will return the disk with additional puzzles for your enjoyment.

Many of the authors mentioned in this article have not been heard from in a long time. If you know of their whereabouts, please have them contact me for ideas about possible future High Resolution articles.

I have only touched on a few of the many programs for the High Resolution board that are available in the Public Domain. There are graphic utilities, patches, enhancements as well as an assort-

ment of un-documented features of the Radio Shack High Resolution Board available. Many of the Programs mentioned this month's article are SHAREWARE and I urge you to support the authors if you choose to use the program. Future articles on High Resolution Software will appear in TRS-Times if readers demand it. Write and let me know if you want to see more. These programs and several others can be found on various BBS systems as well as GEnie and of course THE FILE CABINET'S new High Resolution Program catalog.

STOP THE PRESSES!!!!!

David Huelsmann has just released XARC4 version 02.00.01. This new version will now handle MS-DOS created PKARC files! Several new features have been added including a routine that tells you just how long it took to un-ARC a file. David is hard at work on an upgrade to ARC4 and I can hardly wait to see what he comes up with. I'll keep you posted!

MORE GOODIES FOR YOUR TRS-80

Get the latest issue of TRSLINK

TRSLINK is the new disk-based magazine dedicated to providing continuing information for the TRS-80. A new issue is published monthly, featuring Public Domain programs, "Shareware", articles, hints & tips, nationwide ads, letters, and more.

TRSLINK can be obtained from your local TRS-80 BBS, or you can download it directly from:

8/n/1 #4

215 848-5728

(Philadelphia, PA.)

Sysop: Luis Garcia-Barrio

Believe it or not:

TRSLINK is FREE

DO YOU OWN A TRS-80 COMPUTER AND ENJOY PUBLIC DOMAIN SOFTWARE?

Like most TRS-80 owners you probably enjoy finding new software for your computer but hate to turn over half your paycheck to the Phone Company because you spent all those long hours calling Out of State Bulletin Board Systems in search of new Public Domain Software only to find that the selection was pretty bleak.

THE FILE CABINET would like to suggest an alternative to those long hours and HUGE Phone Bills...

DOWNLOAD THROUGH THE MAIL!!!

Over the years, THE FILE CABINET has collected TRS-80 Software from all over the country. The Programs have been checked, sorted, and cataloged into the largest collection of Public Domain Software you will find. You can enjoy TRANSMISSION ERROR FREE programs without the frustration of long file transfers or dropped carriers. Best of all...you save money by not having to pay Ma Bell an arm and a leg for long distance charges!

Each disk in the library is filled to near capacity with quality software for your TRS-80 that would take months to collect IF YOU COULD FIND THEM!

THE FILE CABINET'S disk library is seaperated into catagories such as UTILITIES, GAMES, EDUCATION, BUSINESS, and COMMUNICATION. There is now a HIGH RESOLUTION catalog available with a READMAC picture file catalog in the works.

SO WHAT ARE YOU WAITING FOR?

A two disk catalog of TRS-80 Model 4 Public Domain Software is available for only $5.00 which is refundable with your first order. The HIGH RESOLUTION catalog is available for $4.00 which includes the READMAC catalog.

Send your catalog requests to:

THE FILE CABINET
P.O. Box 4295
San Fernando, Ca. 91342

Please mention where you saw this ad!

The File Cabinet
Public Domain Software For Your TRS-80
P.O. Box 4295, San Fernando, Ca. 91342

SAVE MONEY... DOWNLOAD THROUGH THE MAIL!

Jack's Tip Sheet

Hardware by Jack Eich

While installing an XLR8er board in my Model 4P, I found that the board easily falls out of the modem slots.

I remedied this by stripping the insulation off of a piece of Teflon insulated stranded wire (.080" OD, 16 gage). I used 2 pieces, each 3 1/2" long, of the insulation, tucking a piece into each of the two plastic guides and centering each in the modem slots.

The XLR8er fits fine now, won't fall out and the added strips stay firmly in place.

While working up a program or reviewing data, I've often wished I could look back at a screen full of information. I've found a way!

In the 4P gate array system, U82 is a 6116 static RAM which contains the information presented on the screen. Pin 18 on that chip is the CE* (chip enable) signal and is wired to ground in order to keep the chip always working. I piggy-backed another 6116 soldering all corresponding pins together except pin 18 (also lifting pin 18 of the original out of its socket). I then mounted an SPDT switch (a tiny one) in the case just below the RESET switch. There is room enough but you must measure carefully after removing the computer face. Two wires run to the two pins 18 and a third from the throw connection of the switch goes to ground on the computer board. I presume you understand the two wires came from the outer poles of the switch. It works fine!

If you want the "second" screen to clear on bootup, (assuming a warm system), as soon as the system clears one screen, throw the switch and press the reset switch again. Now you can start working. When you have a few lines or a screenful that you'd like to refer to later, throw the switch and keep on. Remember, though, that this does not reset the line or character count, so you'll be starting at the spot you left the first screen; but it will scroll, etc, normally. I should mention that each of the two pin 18's will need a pullup resistor to +5V. A 5K ohm, 1/10th or quarter watter to a +5V trace nearby will do fine.

The switch I used has a center position (off). I thought (without really thinking, of course, it was the only switch I had handy), this would be fine. I could use this position to blank the screen when I walk off and let the machine run. **Not So!** The screen fills with junk! Investigating the board I found that I could use either of two unused gates of U92, a 74LS00 NAND gate IC (2 highs = a low).

So, I've run wires from each of the pin 18's of the two 6116's (remember they will be pulled up if the switch is off) to pins 4 and 5 of U92. Pin 6 will go to a third 6116 pin 18 (it won't need a pullup resistor) and this should give me three screens with two to save. I haven't mounted that third 6116 yet because I first must check the clearance. I may have to eliminate the socket.3.

I'm a fanatic when it comes to working on circuit boards. I run my wires on the solder side of the boards. When connecting to socket or IC pins, I desolder (using small wirewick) the pin and straighten it (Radio Shack bends them to hold things in place when wave soldering). When attaching a wire to a pin, I make a very small loop by taking a turn around a small needle, cutting off more than one turn and looping the wire around the pin. I hold the wire in place with a small lump of auto body putty. The putty stays soft forever; I've used the same lumps for years. It's plastic and sticky enough that when pressed over a wire it holds it in place adequately.

Of course I remove it after it's served its purpose. I'm fortunate enough to still have some of the very, very small hollow rivets we used to use to repair circuit boards. When you need to "surface" a wire (bring a connection to the top of a board), I use one of these rivets. The ones I use have an OD of .060" and smaller with a flange on one end and should be only long enough to just come through the board. Hold your board up to a bright light and you can find places where you can install a rivet and miss the traces on both sides. Drill a hold just large enough to accept the rivet. Take a larger drill and with your finders twist it to break the sharp edge of the hole on the flange or "head" side of the rivet. On the side where you want to expand the rivet, make a small depression with the drill. Insert the rivet from the other side, back it up with the flat end of a large drill bit (or rod) held in a vise and tap it with a center punch. It's now permanently in place and neat. Run your wire (I use wire-wrap wire) to the hole, strip the insulation, pull it tight and bend it to hold it in place.

For a quick disconnect, I use undamaged pins from damaged "machined pin" or "high reliability" sockets. Each of these pins have an excellent male end, especially the gold plated ones. (I think all the female ends have gold plated fingers). You can stick the male end in the rivet I mentioned above and solder it in. Don't let any solder get in the female end. Now strip the insulation from the end of a piece of small stranded wire, about 1/16" will do. Insert it into the female end of another socketed pin and solder it. Slip a piece of shrink tubing (aout 1/10" OD) over the wire and all but the male end of the pin. Shrink it and you now have a subminiature disconnect.

DATE YOUR TRSDOS 6.2. THROUGH 1999

Timothy Sewell reviews T62DOSXT

I like TRSDOS 6.2.

It's not that I DON'T like LS-DOS 6.3, I'm just not too sure of it yet. Yes, I bought it; yes I have applied the 5 upgrade patches that were announced over the last year (if you are not up to level L+ you're not current), and yes I am concerned over the alleged protection scheme going haywire (remember the 5 upgrades?).

So I choose to continue to use TRSDOS 6.2.1 and am quite happy to keep on doing so until I am sure of an alternative (the thought of having to convert well over 2000 diskettes as well as a hard drive doesn't appeal to me very much either.) Call me lazy!

Like many of us who decided to hold off switching, we were faced with the fact that our computer would no longer accept a date after 1987. This was no big problem to me, I simply turned the date prompt off and continued to go on my happy way doing what I do with the computer, not caring a bit if the file I have has a current date or not. But, the date is very important to others, primarily those who do programming and need to keep track of their work. So the challenge went out to create the ideal environment for TRSDOS 6.2 that would give us the extended date capability without having to do a major overhaul of our entire disk library. Many have tried and many have come VERY close.

The latest contender to the TRSDOS 6.2 extention challenge is a package by David Goben called T62DOSXT. David has provided a set of patches and utilities that allows TRSDOS 6.2 to accept dates up to 1999 WITHOUT having to do major conversion work on your disk library.

The unique feature of T62DOSXT is the fact that it allows your disks to read and write BOTH TRSDOS 6.2 and LS-DOS 6.3 dating formats. Disks formatted under T62DOSXT have the time stamp that is read by 6.3 but is invisible to 6.2. This means that if you decide to use 6.3 later on down the line, your disks will already be compatible with the date/time stamp format that 6.3 uses.

The patches also add the extended features of 6.3 such as the new SVC routines and continued JCL processing. It also allows you to enter your date without having to add a zero to single digit months and days (06/03/88 for June 3, 1988). Dates can now be entered as single digit entries (6/3/88) which can save a bit of time upon booting up. In fact any non-alpha-numeric character can be used as a seperator for the date entry. You can just as easily use 6.3.88 or 6+3+88, or whatever you feel comfortable with. The choice is up to you. Nice touch.

T62DOSXT also fixes the DIR command so that the directory will be displayed by scrolling onto the screen without first issuing the screen clear command. I have always objected to visually losing what I was doing when needing to look at the directory. Now I have it the way I like it.

Programmers will appreciate the fact that two supervisory calls, otherwise only found in LS-DOS 6.3., have been added. These are the @HEXD and @SCRNPRT SVC's.

OPTIONAL patches include the ability to boot up in all-caps, disabling passwords, dumping below 3000H, speed up of MEMDISK, and scroll protect of up to 15 lines.

T62DOSXT is easy to install. A JCL file is provided that will give you two system disks when finished. The structure is very similar to the one that was used to upgrade TRSDOS 6.2.0 to 6.2.1. The prompts are very easy to follow and you will have no problem following through it's routines.

Included on the disk are some utilities that will make life easier for some:

SYSDRV/CMD allows another disk to be the system without changing logical drives.

ONEPASS/CMD is a fast single pass back up program. You will never again need to FORMAT a disk prior to BACKUP.

NEWDOS/CMD allows you to copy programs from Newdos/80 single sided disks to TRSDOS 6.x. format.

UTILITY4/CMD is the latest version of David's multi purpose disk zapping program. This program is a must for any serious TRS-80 user.

The disk also contains a set of utilities that will allow you to convert older format disks to the extended dating format, as well as un-converting back to the original older format. Now you can have your cake and eat it too.

The package is pretty complete and answers to questions can be found in it's many documentation files. T62DOSXT is a fine conversion package and should be considered by serious TRSDOS 6.2 users who just aren't ready to take the plunge into LS-DOS 6.3.

T62DOSXT is NOT Shareware. It is a program that is distributed and supported by David Goben, 28 Monticello, Willimantic, CT 06226.

The price is $18.00 and is well worth it.

ITEMS OF INTEREST

HARDWARE

XLR8 BOARDS

WITH memory chips AND docs.

$225.00 (only 4 left)

LNW DOUBLERS

without 1791 chip for
LNW & Model I.

$10.00 (30 available)

MODEL I DOUBLERS

NEW
- in original box -
NEVER USED
with binder of docs. RS#26-1143

$25.00 (2 available)

AZTEC 65 WATT POWER SUPPLIES

NEW - in easily removable shield case.
Originally used in Model 4/4P until Radio Shack substituted them with cheaples.

$20.00 (many available)

Please add $5.00 shipping & handling for each item.

JACK EICH
1643 BOLINGRIDGE DR.
ORANGE, CA. 92665
(714) 637-2943

SOFTWARE

* * * A N N O U N C I N G * * *

| T62DOSXT |

Take TRSDOS 6.2 (c) to the year 2000 with the only set of patches that don't play games with your directory!

Tired of kluge patches? Don't want to have to convert even your archival data disks in order to read them properly? Don't want to go through a long, drawn-out process to upgrade your system disks?

T62DOSXT is your answer. T62DOSXT provides 100% compatability with BOTH the old dating standard AND the new LS-DOS 6.3/LDOS 5.3 dating standard. It will read and write to both types of disks with equal, and COMPLETE SAFETY.

ENHANCED Features provided by T62DOSXT

* Fully supported dating from January 1, 1980 through December 31, 1999.
* A relaxed date/time entry on boot-up or when using DATE/TIME. No more need for leading zeros.
* The problem of a running JCL file aborting when it executes a SYSTEM (SYSTEM=d) command has been eliminated.
* Screen clear during directory listing paging has been removed.
* Support of the new @HEXD and @VDPRT SVCes for 100% program portability with LS-DOS 6.3.
* Numerous OPTIONAL system patches are included, such as disabling passwords, boot-up in all-caps, faster MEMDisk formatting, and allowing up to 15 lines of Scroll Protect.

Included in T62DOSXT is NEWDOS; a NEWDOS80-type CONV utility, SYSDRV; change system disks WITHOUT changing logical drive names, ONEPASS; a single-pass disk format and backup utility, UTILITY4; a disk/file zapper, DATECONV; convert the disks you need for compatability with the old AND new dating standards, and UNDATE; this utility allows you to convert a disk BACK to the old standard.

ORDERING INFORMATION

T62DOSXT is only $18 (US funds), plus $2 shipping and handling for US destinations. Canada, Mexico, APO, FPO, and US Territories with US zip codes add $3 S&H. $7 S&H anywhere else. Send your order to:

David Goben
28 Monticello, Willimantic, CT 06226

Dave, please send me _____ T62DOSXT disks. That totals $_____, plus $_____ postage and handling (NOTE: orders for up to 3 carry the same postage as 1). Enclosed you will find $_____ to cover the goods, so quit reading this and fill my order. (Sorry, no charge numbers or purchase orders.)

Your Name _____

Address _____ Apt # _____

City _____ State _____ Zip _____

Orders shipped next work day after receipt. Same day processing.

(NOTICE: TRSDOS is a Tandy Corporation tradmark. LDOS and LS-DOS are tradmarks of MISOSYS, Inc. T62DOSXT is a COMMERCIAL program, and is supplied on a DATA disk. TWO disk drives are required for upgrading your TRSDOS 6.2 system.)

TRS-80 SOFTWARE for Models 1/3/4/4P/4D

Many useful programs - Economical prices
Send $2.00 for listing

PRACTICAL PROGRAMS
1104 ASPEN DRIVE
TOMS RIVER, N.J. 08753

BULLETIN BOARDS

• Colorado
• Denver

KINGS MARKET
(303) 694-9020
300/1200 baud (24 hrs)
Sysop: Jim Burt
Supports: Model I,III,4, CoCo, Orch. 90 & CP/M

TBBS ISLAND
(303) 690-9423
300/1200 baud (24 hrs)
Sysop: Mike Hastings
Supports: Model I,III,4,100, CoCo & Tandy 1000

• Longmont

DINO BBS SYSTEM
(303) 652-3595
300/1200 baud (24 hrs)
Sysop: Chris Anderson
Supports: Model I,III,4 & CP/M

T.B.B.S. (R)emote
(303) 678-1469
300/1200 baud
(9:00 am - 8:00 pm MST - everyday)
Sysop: Danny Peterson
Supprots: Model III & 4
Use 2 <CR> to log on.

PUBLICATIONS

CODEWORKS

This bi-monthly magazine comes from the folks who used to publish the TRS-80 specific publication, 80-US, later known as BASIC COMPUTING.

CODEWORKS is not TRS-80 specific, instead the focus is on Microsoft GW-BASIC. However, each and every program listing has the translations to Model I,III & 4 Basic when needed.

This is a fine magazine with lots of information, tutorials and Basic type-in programs.

The subscription rate for 1988 is $24.95.

Contact:

CODEWORKS
3838 South Warner Street
Tacoma, Washington 98409

*** NEW ***
Recreational & Educational Computing

Have you been missing out on the only publication devoted to the playful connection of computers and math?

The REC Newsletter features programming challenges and recreational math, such as: The Magic of Schram, 123 String, The probability of an N game at Bingo, time to complete a collection, 6174, Next Number In Sequence, Locate the Bomb, perfect numbers, Fibonacci numbers, prime number generation and contest, self reference and paradoxes, self listing program challenge and solution, pi, mystery programs explained, probability, Monte Carlo simulations. Also: Fractal art, the world's best card trick (based on algebra), reviews of best software and books, editorial, humor, cartoons, art, reader solutions, and more!

Programs supported for TRS-80, Tandy, MS-DOS and others.

REC is available for $24.00 per calendar year of 8 issues

REC Newsletter
129 Carol Drive
Clarks Summit, PA. 18411
(717) 586-2784

SUPPORT for your TRS-80

THE ONLY MONTHLY PUBLICATION
THAT SUPPORTS YOUR
MODEL I, III, IV, 4P & 4D

CONCENTRATION IS ON THE USER APPLICATION OF PROGRAMS, SOURCES OF PRODUCTS, PRODUCT REVIEWS, FEED BACK LOOP AND NEWS ITEMS FOR THE TRS-80 USER

$ 18.00 FOR FIRST YEAR INTRODUCTORY OFFER MAILED IN THE US
$ 29.50 CANADA AND MEXICO; $ 30.00 OUTSIDE THE US, CANADA AND MEXICO

Sample Issue $2.00 US And Canada

307-265-6483

Computer News 80

P. O. Box 680
Casper, Wyoming 82602-0680

CLOSE#5

This issue of TRSTimes was tough putting together. The copy you are reading now is version 7. That is, it is the seventh different layout of the September issue. The problem we had was the kind that every editor dreams of: More material than alloted page space! There were just so many interesting articles that it became difficult to choose the ones that would create what we consider to be the right balance of information. I hope we made the right choices.

In the process several fine submissions were not included, such as Fred Cornet's Visicalc utility, Robert Doerr's demonstration on 'Windows in Basic' and Jim King's plotting program. This material will appear in upcoming issues. Also, you might have noticed that 'Hunting for Buried Treasure' is missing from this issue. No, this series has not been discontinued. There just wasn't room. This particular installment will be shuffled to a spot next year.

Meanwhile, to David Goben, who took time out from his busy schedule with Computer News 80, to give us nice enhancements to Scripsit .....

To Jack Elch, who keeps most of our machines running, for sharing some hardware tips.....

To Fred Blechman, whose numerous articles we have read in every conceivable computer magazine, for stepping us through the sale of an unwanted computer.....

To Tim Sewell, the Guru of all sysops, for his review of T62DOSXT, available hi-rez programs and the layout of the front cover....

To Roy Beck, our thoughtful teacher of CP/M, for bringing coverage of this neglected Model 4 DOS.....

To Barbara Beck, whose attendance adds much needed class to the various computer club meetings, for telling us what Roy is REALLY like....

....a great, big THANK YOU from TRSTimes.

A very special thank you goes out to Gordon Collins and Ruud Broers.

Gordon is the editor of the National Tandy & General User Group newsletter from England, called NATGUG News. His article on SYS13/SYS is indicative of the information you can expect from this fine publication which we recommend highly.

Ruud Broers gives us a nice Newdos/80 utility called MACROKEY. Let's think about this for a minute! Ruud, with editing help from Peter Plomp, wrote this article and program. They are Dutch - from Holland - English is not their native language. Pretty impressive. When was the last time any of us wrote an article or program in Dutch? We have trouble enough expressing ourselves in our own language. Think how difficult it would be to program in Basic or Assembly Language if the keywords and mnemonics were in Dutch.

With the help of all these very talented people we have produced what we consider to be our best issue to date.

Next issue, November, will bring some of the articles we did not have room for this time. 'Hunting for Buried Treasure' will be back with a monster-size installment featuring a joint venture by Tim and Lance. George Madison has promised us his update to TRSTEXT, the program that allows the use of DOTWRITER fonts on the high resolution screen. He calls it TRSTEXT2. We have seen it and, boy, it is fast. The PD Express and CP/M columns will be back with more good stuff so,

STAY TUNED - DON'T TOUCH THAT DIAL!!

ATTENTION MODEL III USERS

Some weeks ago Norm Freidin sent us a flier which he had received from a Canadian company called DBSIDE. The flier advertised an enhancement to TRSDOS 1.3. that would allow double sided access.

"Is this on the level?", Norm asked, so we contacted DBSIDE. A few days ago we received the full package.

The answer to the question is: **YES, INDEED!!**

It does exactly what the advertisement promised: it allows you to use both sides of your disks, something that TRSDOS 1.3. was never able to do previously. Of course, you must have double sided drives to take advantage of this.

DBSIDE is not just a series of patches. It is a complete machine language program that rewrites large portions of DOS to allow double sided access. In the process it also enhances several existing commands and utilities, as well as adding some new library commands to make your life easier.

This is a GOOD program and the cost of $29.95 is very reasonable. If you use TRSDOS 1.3. regularly, DBSIDE is certainly well worth thinking about.

For further information contact:

Gary Campbell
c/o DBSIDE
Suite 209 - 1051 KLO Road
Kelowna, British Columbia
Canada V1Y 4X6

One final note: When you purchase a product from a vendor, tell them where you heard about it.

If you read about it in CN80 or TRSLINK, tell them so. Of course, if you learned about the product in TRSTimes, be SURE to tell them.

Until November, as so eloquently stated by Tim,

"Make mine TRS-80."