

# THE S-BIGHTY™

SEPTEMBER 1980 • FIFTY CENTS

## BACK TO SCHOOL WITH THE MICROCOMPUTER



# INDEX

<b>Feedback</b>	4
<b>Editorial</b>	5
<b>New Products</b>	6
<b>Radio Shack's "Versa File" - A Review</b> by A. A. Wicks	14
<b>Reader Service Information</b>	16
<b>Speed on the TRS-80</b> by James H. Sheats	18
<b>Opposite Graphic Relationships</b> by Stevenson Dishcert & Keen	20
<b>Back to School</b> by Sherry Taylor	22

Dear Sir:

I must admit that I was very surprised when I read the Letter to the Editor in the May 1980 issue of your publication. As a computer programmer, and as a person who believes very strongly in good business ethics, I feel I must respond to the letter that Mr. Hern wrote.

Apparently, Mr. Hern, you are not a computer programmer. If you were, you would understand the amount of time, energy, and just plain hard work that goes into software development. When I talk about software, I am speaking of more than just the "games we enjoy", although they are of equal importance. I am talking about very large business applications, complex utilities, compilers, interpreters, languages, and operating systems. Some of these programs can literally take years to design and write. Consider this example:

For the past two and one-half years I have been writing a disk operating system (DOS). Writing a program of this nature is a very taxing process, requiring hours of research and experimentation. Just when you think you have a routine perfected the computer bombs out during a critical moment because of a tiny bug, which you spend hours tracing. After two and one-half years of work, the task is only about one-third complete. I am still not satisfied with the completed work. After all the other work is finished, I can devote only about ten hours a week to the project.

Let's take a look at the economic facts of life. The DOS will take four years to complete. This means that I will not have a marketable product until the end of 1981. At ten hours a week, 200 weeks, and \$8.00 an hour, the total labor bill alone is \$16,000.00. Add to that the figure research fees (not everybody can write a DOS), attorneys costs (I license all my programs), advertising and marketing costs, and the cost of

# FEEDBACK

supplies and the bill is now hovering at \$20,000.00. Since the DOS must be able to run on the large computers (two and three drives instead of one; don't forget the printer.) there are certain hardware costs to consider. Add \$2,000.00. The total production cost for the DOS stands at approximately \$22,000.00. Of course not all of this is spent money; I could "donate" the labor.

Now we have to sell the final results of my four year project. If the DOS is a success, if the buyers recommend the product to other computer owners who then buy their own copy, if someone doesn't market something similar before I release my own, only then do I make my first dollar. Let's just suppose for a minute that my DOS is a smashing success and I receive orders for 10,000 copies. To make up for the money spent, I have to charge \$2.20. The diskette itself is going to run you \$8.00 (given inflation for another year). Shipping, handling, etc. is going to cost \$2.50. I have to make a profit and the DOS hasn't made any money for four years so add \$3.80 (just a tad over 30% of the accumulated costs thus far). Don't forget more legal fees for license processing (\$2.50), and having the documentation printed isn't cheap (\$3.00). The total unit cost to purchase the new DOS is \$22.00 (plus sales tax if you live in Michigan). Remember, also, that you have to wait 30 days before it comes via UPS or Parcel Post.

If I were a dishonest person who wanted a copy of that DOS, and knew someone who had a copy, I could obtain it for the cost of the material (\$8.00). That is a savings of \$14.00. So much for your mass marketing/low cost theory.

The example I cited above took "only" four years to write. Lance Micklus said that Randy Cooke's DOS took SEVEN years

continued on page 19

Publisher  
Roger Robitaille, Sr.  
Editor  
William F. Gollan  
Advertising Manager  
Mary Reed  
Creative Director  
Elaine Cheever  
Graphic Artist  
Donna Bennett  
Typography  
Lauri Miller  
Lester Anderson  
Brian Berkebile  
Diana Bishop  
George Blank  
Richard Bouchard  
Joseph Breton  
Suzanne Breton  
Philip Brown  
Clit Campbell  
Nancy Chase  
Brenda Cookinham  
Donna Cookinham  
Pam Demmons  
Mary Edwards  
Linda Fedas  
Phoebe Fensterman  
Karen Fissette  
James Garon  
Donna Jean  
Bette Keenan  
Bea Kimball  
Rich Koch  
Kathy Maloof  
Thomas Marshall  
Donna McMahon  
Lance Micklus  
Doris Miller  
Glen Ohlund  
Mark Ohlund  
Mark Pelczarski  
David Robitaille  
Elizabeth Robitaille  
Eric Skyten  
Richard Taylor  
Jack Torres  
Anmar Williams  
Gary Young

The S-Eighty is published monthly by SoftSide Publications, 6 South Street, Milford, New Hampshire, 03055. Telephone (603) 673-5144. Subscription rates: USA Bulk Rate, \$5.00 per year. All remittances must be U.S. Funds, first class and overseas rates available on request. Mail all subscription inquiries to: The S-Eighty, SoftSide Publications, P.O. Box 68, Milford, New Hampshire 03055. Entire contents copyright 1980 by SoftSide Publications. All rights reserved.



The S-Eighty  
A SoftSide Publication  
P.O. Box 68  
Milford, NH 03055  
603-673-5144

\* Please note: Radio Shack and TRS-80 are registered trademarks of the Radio Shack Division of Tandy Corporation.

I'm concerned. I'm concerned for the thousands of ordinary folks out there who are being duped into believing that they can make millions of dollars in the "fastest growing field in the nation" simply by writing software for micros. Don't get me wrong; there certainly is money to be made . . . but millions? Come on, let's be serious for a minute. Do you personally know any programmer who has made 'millions'? Frankly, I doubt it. To tell you the truth, I can't think of one. Thousands perhaps . . . but millions? No way.

How many young singers drift away with dreams of riches and bright lights only to come crashing back to earth, shattered? Just take a minute and list some of your favorite pop music stars. How many could you name . . . fifteen, twenty perhaps, thirty at most. Well, whatever the number may be, I'll bet that less than 30% of those rock'n rollers ever made a million bucks, not to mention the tens of thousands who never made a dime. "What does all that have to do with me", you say? Well, I'll tell you.

The parallels between microcomputing and the recording industry are uncanny, and in time we will have a few "super stars" of our own. Be patient; it won't happen overnight, and it won't happen without practice. Some day we may have our own 'Top Forty', and with that day will come the appropriate national recognition that our 'Super Star' artists deserve. But some people would have you believe that you'd be driving by the Los Angeles Coliseum, only to see swarms of packed micro-groupies anxiously awaiting the arrival of their 'Micro Star'. A scene out of a science fiction comic perhaps, but until we have the fans that the recording stars of today enjoy, the odds of our ever finding individual millionaire programmers are mighty, mighty slim.

As Project Manager at Kilobaud/Instant Software, I occasionally read an editorial or two. The one that I remember the most was June '79 Kilobaud, part of which appears below.

### Getting Rich

Would you truly like to be rich? Would you like to own your own plane, perhaps a yacht and maybe an Arabian horse? How about being able to travel anywhere in the world you want . . . perhaps a 30-room home on 50 acres with a fantastic view, your own fish pond and even an outside sauna? Fantasy? Not one bit of it. There are fortunes to be made in microcomputing if you go about it the right way. No matter what your education or intelligence, you can become a millionaire . . . if you really want to.

Anyone who really wants to can become a millionaire within five to seven years - and in the microcomputing field I'd shorten that to a maximum of five years, just starting from scratch.

It's been a year or so since I left Kilobaud, and you know I've yet to find all those millionaire programmers. Can you help me?

# RADIO SHACK'S "VERSA-FILE"

As a person with a great many diversified interests, I am always jotting down notes or reminders regarding a multitude of subjects. When the volume of these notes gets to be a little unwieldy, I review them -- and frequently end up with sub-notes!

When I first saw the announcement by Radio Shack about a disk program called "VERSA-FILE", it looked like a way to finally get organized -- but still keep my note "system". Indeed, it has permitted me to do that, and more. Almost daily I find a new use for this unique program.

"VERSA-FILE", in effect, permits you to keep note records of any information chosen, with instant recall of this information by an input query. For instance, you might place this statement on file: "Passport is in security box top shelf of bedroom closet." (No quotation marks are used in actual input -- just the sentence with a period.) Weeks, or months later, you might wonder where you put your passport for safekeeping. You do remember that you had listed it in "VERSA-FILE". So you query the file with "Where is my passport?", and promptly the answer comes back: Passport is in....etc. The possibilities of keeping track of seldom-used or easily-forgotten items or documents, "etc". is endless. There would appear to be some reason to consider the program for inventory purposes, too. However, I feel that the entry mode (individual statements to disk) would be unwieldy, and, as there is no sort capability, there are more efficient programs that are available for inventories.

As an active amateur radio station operator, I frequently wish to know the date when I contacted a certain station. Although I keep a log, it is necessary to scan the "call" column for months back in order to determine this. Now, with "VERSA-FILE", I enter all calls for a particular day with the statement: "Contacts for 06/03/80 were: W6BRG, KA6YFA, YU4HO....etc." Then, when I need a contact date, I query, "When was contact with W6BRG?", for example. The reply provides my original input statement, which, of course, includes the other station calls, too. I may also ask, "What contacts were made on 06/03/80?", and get the same information.

Another application with some good possibilities is language translation training, similar to that now done on the handheld translators, or just for study purposes. For example, enter "In German, Good Day (or 'hello') is Guten Tag." Then, you can query, "What is Good Day in German?" -- and, sure enough, there is the answer! History, politics, dates, mathematics can all be similarly handled.

What is happening here, and how does Radio Shack's program author accomplish this? Well, without going into the deep details of the program

itself for its technicalities, the basis is Keywords, and each Keyword is a data file. The Keywords on this disk program are: "is", "are", "was", "were", "will", "has", "can", and "the". However, the program is in BASIC and may be listed and edited, if you wish. Therefore, you may change any Keyword to suit your needs. As an information entry is made, the data is stored within the Keyword file. That means, of course, that if you enter a statement that contains no Keywords (and that's hard to do!), then the statement will be placed in the last file ("the") along with any "the" entries.

At the time a query is given to the computer, a search is made within the related file for a statement containing companion words to the query. As some statements on file could be closely related, more than one statement could be returned. For instance, if, in our passport example given, we had entered an additional statement, which said that Bob's passport was in the desk drawer, and then asked "Where is the passport?", both passport statements would appear. But by making the query more specific, the search is restricted and will provide the single answer desired.

If you are not sure in which file one of your entries resides, you can perform a "Global Search". In this case, if you provide the word that you are searching for (again, for example, "passport"), then the response will be all statements containing the word "passport". This will occur even though the original statements may be in different Keyword files. Similarly, an entire file may be produced by merely typing the Keyword and a question mark (e.g. "is?"). At any time, if you do something unacceptable to the program, the response will be, "I don't understand your entry. Please try again."

Any or all files may be "killed" at any time, either individually, or as a "multiple kill". A warning is issued in the latter mode, which states that all records in that file are about to be killed. Upon completion of any kill action, a statement is displayed, advising the number of records killed. On a multiple kill, you have a last chance when you are asked, "Are the records kills correct?" Also, in a multiple kill, the records must all be in the same file; otherwise the display will interrupt your input and state, "Not in same file!" In this case, a new kill must be initiated for the "foreign" Keyword file.

The manual for the program is contained in a "TRS-80 Model I" brown plastic three-ring binder of good quality. One clear, plastic, two-diskette sleeve is provided for the supplied diskette, and, presumably, a backup diskette. The manual itself is extremely clear and would permit anyone who

continued on next page



continued from previous page

has no knowledge of programming whatsoever, and only a rudimentary knowledge of TRS-80 operation, to run this program without any difficulty. Many examples of each function are given (something that is needed in many program guides). For those demanding a more thorough background, a call for "List" will give them the whole program, which, incidentally, is fairly complex.

The excellent printing and layout of the manual is somewhat marred by several spelling errors and some typographical mistakes, but these are minimal. An Appendix provides information on formatting disks and on making backup copies, but fails to address the poor single drive owner.

Printing is supported on any question or answer. Minimum system requirements are 16K Level II computer, Expansion Interface, and one Disk Drive.

As a very useful information filing program, "VERSA-FILE" meets all the requirements, and more; its possible varieties of use could be considerable. The price of \$29.95 is certainly reasonable, if you can utilize its full capabilities. But unless you can, then the cost to you will be high for something that is just "played" with and then put aside. Hopefully, this rather detailed review will help you to make the correct determination.

VERSA-FILE. Catalog No. 26-1604, by Radio Shack, a Division of Tandy Corporation. 

### CASSETTE BASED MACHINE LANGUAGE PROGRAMMERS, REJOICE!

Are you tired of reloading Radio Shack's Editor/Assembler and the source and object tapes every time you want to try out a program that you are debugging? If so,

**YOU CANNOT AFFORD TO BE WITHOUT**

## "ASPTCH 2.0"

by James F. Williams

**ASPTCH loads after EDTASM and adds these functions:**

- |  |  |
|--|--|
| 1. Reserve area at top of memory   | 6. Enter Level II BASIC's Monitor mode (system prompt)   |
| 2. Display number of bytes left in text buffer                             | 7. Enter Level II BASIC's command mode and use all functions that do not use variables   |
| 3. Dump assembled program DIRECTLY into that area without using a cassette | 8. Return to fully programmable BASIC, keeping dumped programs intact, if desired (for use with BASIC's USR command, for example). |
| 4. Execute the dumped program  |  |
| 5. Convert and display memory locations and their contents.                |  |

Included with ASPTCH 2.0 are two utility programs: SYSTPE which permits you to copy any number of distinct memory areas into one System format tape and BREAK, which allows you to halt execution of a program being debugged and display flag and register conditions.

ASPTCH requires TRS-80 Level II, 16K and up and is used with Radio Shack's Editor/Assembler 1.2

All three programs only \$19.95 with complete instruction manual and cassette (Instruction manual only, available for \$1.25)

**ORGAN:** Play your TRS-80 like an electronic organ. Video display shows a two manual keyboard with corresponding TRS-80 characters overlaid. You can play single notes or chords, change sound quality, select from two loudness settings. Output sound to cassette or directly to a small amplifier through the auxiliary jack. Instructions and program cassette only. \$14.95

## MICROPUTE SOFTWARE

"WE DO ALOT WITH A LITTLE COMPUTER"

**HOUSEHOLD FINANCE PACKAGE:** Plan and monitor your spending with this budget program. You can balance your checkbook, record and print savings information, group budget items for sub-totals, monitor credit accounts, investments, and tax deductions. For Level II, 16K cassette and up or 32K disk and up. SPECIFY DISK or CASSETTE version. Programs and instruction manual. \$29.95

**DISK LIBRARY:** A two program system that provides simple indexing of all your diskette programs. Individual disk libraries recorded on each diskette permits the access of any program by simple number entry as displayed on video output. The program also provides indexed hard copy printouts of disk program library. Editing feature allows for adding, deleting, or correcting library information. Two programs and instructions. \$9.95.

**CHEMISTRY I:** Laboratory workers, students and researchers can solve the fundamental laboratory problems that taunt you daily. Program solves solution mixing problems (Molarity, Normality, Molality), titration problems, ideal gas law, pH, temperature and metric conversions. Level II, 16K and up CASSETTE version \$29.95, disk version with file storage capability \$37.50. Programs come with complete instructions. (Chemistry II will be released in Oct.-Nov. 1980)

**CONTRACT PROGRAMMING:** Contract programming is available in business, science, and computer assisted instruction (CAI) areas. Please send inquiries to the address below. Briefly describe needs and specifications for estimate on contract work.

**TERMS:** We accept check, money order, Master Charge or Visa Card. North Carolina residents include 4% sales tax. Please enclose \$1.00 shipping and handling charges on all orders.

Send your order to: Micropute Software, P.O. Box 1943, Rocky Mount, North Carolina 27801. READER SERVICE 28

# SPEED ON THE TRS-80

Much has been written about the "speed" of data processing. Certain CPU chips are "fast," there are "fast" and "slow" RAM memory, and "speed of execution" is measured in nanoseconds and microseconds. What does it all mean, and is it relevant to the average TRS-80 owner? As usual, the stock answer is that it depends on the situation.

There are several situations in which speed is of great importance:

1. In process control and other real time situations.
2. In time sharing situations.
3. When you are using leased transmission lines.
4. When utilization approaches full capacity.

In the first situation, the computer is interfaced to a series of measuring instruments and controls so that it can measure certain parameters and take certain corrective actions based on the readings it obtains from the controls. The "speed" of the computer determines when the next measurement can be taken and is very important in industrial quality control, since the process has been unsupervised since the last measurement was taken many nanoseconds ago. If anyone is using the TRS-80 for control applications, he has the material for an excellent article of his own and does not need advice from me.

However, now that Micronet, The Source, Forum 80, and various Community Bulletin Board Services are in operation, the second and third situations will come up more often. Even though the BAUD rate is standardized at 300, it is still possible to create savings in the pocket by cutting down on what is transmitted and by trying to say the same thing with fewer bits.

Situation Four will probably always be the most common case. In its classical textbook form, this is where the programmer attempts to avoid something expensive, like a new shift of operators in the computer room or the purchase of a larger machine, by finding a way to get the required output through the computer in less time. The TRS-80 owner is not generally faced with a decision of this nature, but he probably would like to cut down on long pauses in the program where nothing visible is happening. He would like to get on to the interesting parts.

When is speed not critical?

1. When the program provides for a great deal of interaction between user and computer. The computer's delays, measured in nanoseconds, fade into insignificance as the machine waits for the user to scratch his head, refocus his bifocals, and shift his chair prior to entering his response.

2. When the computer is "peripheral bound." Regardless of how efficient the programming is or how fast the CPU is, if you have a slow printer, you just don't finish the job significantly faster.

Still, attempting to speed up one of your programs is a good intellectual exercise and not without its own set of benefits. Even if you only use your TRS-80 for games, cutting down on the length of time necessary to draw a backgammon or checker board on the screen can increase your enjoyment of your machine, so it's still a worthwhile endeavor. So how do you go about it?

First, learn more about TRS-80 BASIC. There are many alternate ways of coding the same program. Some run faster than others. Most good BASIC textbooks, including the famous Radio Shack User's Manual for Level I, may have hints for speeding up execution. Read several of them. They are rarely totally repetitious, and sometimes if a reader is told the same thing several times, it finally sinks in. After reading several textbooks, read some of the articles in the various hobby journals, including this one. Occasionally some hobbyist full of wisdom will seek to share his own discoveries on speeding up execution with other hobbyists. A trick that I have used with slow programs is to save the original program and then find a part of the program that has a fairly definite starting and stopping place. Time the execution of this part of the program with a stop watch or any timepiece with a second hand, and then begin to make your changes. Retime the program from time-to-time to see if various changes have helped or hindered execution speed. When you have arrived at what you consider to be your best effort, save it before you sign off the computer. You can use it as a starting point, next time.

Second, you might try a different higher level language. Now that FORTRAN, COBOL, FORTH, APL, PASCAL, BASEX, and tiny C are available on the TRS-80, maybe one of these languages will pick up the execution speed of the program. There are also other BASIC dialects, and some of them are compiled rather than interpreted versions, so that you would actually be running a lightning-fast machine language object program.

Next, with increased involvement on the user's part, but probably less expense, you might want to program in Assembly Language, using the Editor-Assembler, or Machine Level Language, using T-BUG, or you can use one of the other monitors. This is, of course, quite involved and should be undertaken only if you mean business.

First try putting machine language subroutines into BASIC programs. Then try whole programs in machine language. Again, the hobby magazines will provide some help.


Mentioned last, for good reason, is the possibility of speeding up execution time by doctoring the inside works of the TRS-80. In the first place, all of the advertising and other discussion of this type of modification indicates that it will

continued on next page

continued from previous page

speed up execution time by approximately fifty percent. Most of the advertised software, including new language compilers, promise speed increases of factors of from three to twenty times the speed of Level II BASIC. Obviously, then, the software route is the way to go before trying anything else. In the second place, these speed-up modifications will probably void the Radio Shack warranty. However, if you have done about all that you can do in the software direction, presumably you will still be able to improve your best effort to-date with a speed-up modification.

So if your wife has become restless watching

you play Star Trek, try spending your time speeding up one of your more practical programs. It is an excellent way to learn more about programming. 

Feedback continued from page 4

to write. If programming is putting bread and butter on your table, seven years is a long time to wait for a project to begin making money.

Above and beyond all else, the law is quite specific with regards to theft. Our society places stiff sanctions on those who steal from others. There are patent and copyright laws to protect literature and inventions. If you want to "hang the ethics" because you feel "no one gives a rat's tail about morality", does that give me the right to go out and shoot someone who steals my programs?

Software that people purchase for their computer belongs to the original purchaser. Let them modify it, rewrite it, destroy it, whatever they like, so long as they don't copy it off for their friends and relations to use. Any fool can take other peoples' programs and append them; that hardly qualifies as writing software.

As for you, Mr. Hern, business people who "swallow their pride" when theft is occurring right underneath their noses are an even greater threat to our free enterprise system than those who do the stealing.

Respectfully yours,  
Christopher C. Gillett  
Software Design Laboratories  
Harrison, MI 48625

My sentiments exactly, Chris. Great letter. I really wonder how many readers actually took Mr. Hern's letter seriously. Frankly, I discounted the entire piece as pure rubbish. On one hand he characterized the "... political left ..." as unappreciative non-producers and, later on, advocated "hang the ethics ...". It sounds to me like Mr. Hern's answer to the copyright infringement problem is quite similar to that which he

accuses the "... dwellers in the dreamland of the political left ..." to be guilty of. By the way, what does the "political left" have to do with software piracy?  
W.F.G.

Dear S-Elghty folks:

I guess if you're just doing a compendium of TRS-80 advertising, you're doing an OK job. The little polemic by Bill York in the June issue entitled "Economics?" looked to me like an eleventh-hour afterthought cranked out at high speed - I can write better in my sleep - and who was it directed at? Certainly not the readers. Maybe we've got a democratic organization here of the advertisers, for the advertisers, and by the advertisers.


But if that's so, why have editorial content at all? As an author, I was (expletive) that you didn't publish the listing of my program ONECLAP last month. And I was also (expletive) that I didn't receive a single review copy. I loaned out my "subscription" copy and the recipient lost it, so now I don't even have a copy of my own article! But I wasn't (expletive) enough to write.

But I was genuinely interested in Scott Snyder's subroutines. His article (after the funny part) says "We must first look to line 110 ...". Like my ONECLAP, you printed the article (written to complement the listing), but not the listing, making the author feel dumb, the reader infuriated, and DOS offended.

Bill York: if you listen to that stuff on the radio going to work, your brain turns to MUSH. Roger Robitaille: if you need someone to put the editorial material together right, I'm available.

Michael Potts

P.S. Tell Scott Snyder I'll trade him a listing of ONECLAP for a listing of his subroutines.

Michael, which came first, the chicken or the egg? It sounds to me like your anger towards us for not running the "ONECLAP" listing has surfaced in the form of a blanket indictment of our publication, which I believe to be unfair. A publisher generates the bulk of his income directly from the pages of each month's issue. The June issue consisted of 53% advertising and 47% editorial. Perhaps to someone 'outside' the publishing business this ration seems excessive. We barely cover our costs with a 50/50 mix of editorial vs advertising. Aren't we entitled to a ham sandwich? I can't fault Bill York on his piece entitled "Economics", other than to say that the S-80 is not an industry newsletter and its' readers are more concerned with end user problems than with those that beset advertisers, perhaps we erred in running it. We'll watch it next time. Thanks for your offer to assist us editorially. We can all use some help from time to time. Well, I must sign off now; I have a terrible headache, probably from listening to all those Boston radio talk shows?  W.F.G.

# OPPOSITE GRAPHIC RELATIONSHIPS

Few people have noticed that there is a direct relationship between a graphic character and its opposite on Radio Shack's Level II computer. By using a simple routine, a graphic character on the screen can be reversed; that is, the lit squares can be darkened and those that are dark can be lit. This can help create some very interesting and eye-catching video displays. That is the application which caused us to first realize that a pattern existed.

We had drawn a picture on the screen which we planned to use as a display in a window. We wanted a scan from the top to the bottom which would change the picture from a "positive" to a "negative". The discovery of a direct relationship between a graphic character and its reverse enabled us to have the animated display we had hoped for.

Here's the relationship. As you know, a graphic character is made up of 6 blocks. ASCII Code 128, no blocks lit, is the reverse of ASCII 191, all blocks lit. Code 129 is the reverse of 190, 128 is opposite 189, and so the pattern goes until it reaches 159 and its reverse, 160. Figures 1 and 2 demonstrate this pattern of the end characters matching opposites on the way toward the middle.

The routine for reversing is shown in Figure 3 from lines 50 to 200. It reverses scanning from the top of the tube to the bottom and then reverses again (Line 90) going from the bottom up as if it bounces off the side of the screen.

The beginning lines simply throw random "garbage" graphics on the screen.

The actual math and logic portion of the program can be described in a nutshell. A video memory address is peeked and a value is obtained. The computer takes that value and subtracts 128 from it. The result is

OPPOSITES

128	191	144	175
129	190	145	174
130	189	146	173
131	188	147	172
132	187	148	171
133	186	149	170
134	185	150	169
135	184	151	168
136	183	152	167
137	182	153	166
138	181	154	165
139	180	155	164
140	179	156	163
141	178	157	162
142	177	158	161
143	176	159	160

FIGURE 2

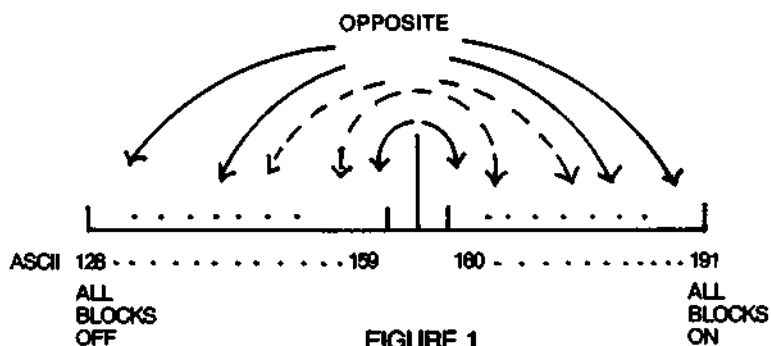


then subtracted from 191. The value gotten from that operation is examined to see if it is greater than 191. If it is, 191 is poked into the same video location; if it's not, the remainder is poked after subtracting 128 from the peeked value and then subtracting that from 191.

The routine won't reverse numbers or letters. Should your screen contain these characters, a line could be added to the routine to tell it not to touch the value stored in that location.

A nice feature about this routine is that it doesn't matter whether the characters have been placed on the screen by a CHR\$, STRING\$, or set statement.

There are the tricks. Just get your imagination into gear and put it to work for you.



10 ' ROUTINE FOR THROWING RANDOM GRAPHIC CHARACTERS ON THE VIDEO

20 DEFINT A-Z:CLS:FOR X=1 TO 400:R=RND(63)+128

30 PRINT ORND(1023),CHR\$(R);

40 NEXT

50 '

ROUTINE FOR REVERSING VIDEO.

60 NN=128:CC=191:FORM=15360 TO 16383:N=PEEK(M):C=N-NN:P=CC-C:

70 IF P>191 THEN POKEM,CC ELSE POKEM,P

80 NEXT

90 NN=128:CC=191:FORM=16383 TO 15360 STEP -1:N=PEEK(M):C=N-NN:P=CC-C:

100 IF P>255 POKEM,CC:NEXT M

110 IF P>228 POKEM,N:NEXT M

120 IF P>191 POKEM,CC:NEXT M

130 POKEM,P:NEXT M

200 GOTO 200

FIGURE 3

# BACK TO SCHOOL

322 South 21st Street  
Haines City, FL 33844

Dear Sir:

In the May 1980 issue of the *S-Eighty* an article by Michael Potts appeared. He seemed upset at the lack of good educational programming. And, in some ways, he's right. It is hard to find. However, if you look hard enough, it's there.

I discovered these two programs, "The Playful Professor" and "The Human Adventure" through an ad in 80 *Micro Computing*. I have been very impressed with them.

I asked a math teacher his opinion of "The Playful Professor." He is very excited at his prospects of using it in his classes next fall.

I have no connection what-so-ever with Med Systems Software. I submit this article to let people know that these programs are available.

Thanks for your consideration of my article.

Sincerely,  
Sherry M. Taylor

For those of you who are bemoaning the fact that there are few educational programs available, Med Systems Software of Chapel Hill, N.C., has brought out two new programs that are worth mentioning.

Most kids really hate drilling with math problems, but "The Playful Professor" makes it a little more fun. With this program two youngsters (or oldsters) can drill together. The program sets up a sixty-room mansion complete with a pesky ghost who has the key to the front door. The player must chase the ghost to get the key and make it back to the front door before the ghost or other player takes the key from them. The doors to the rooms open and close at random with each move.

To make moves through the mansion, the player must answer the math problem correctly. Then the number of moves is determined by the roll of dice. If the player is temporarily trapped in a room, his move is skipped and

play is passed to the other player. If there is only one player, the next problem is served up.

From the educational standpoint, the tutorial routine of the program makes it worth much more than other drill programs. If a problem is missed, the screen is cleared and the program goes into an explanation using a blackboard format familiar to every student. It takes the student step-by-step through to the solution. Even the tutorial is interactive in the case of reducing fractions to their lowest terms. After the explanation on how to reduce fractions, the program asks the student what number will reduce it.

"The Playful Professor" provides tutoring in integer mathematics as well as fractions, for the four basic operations. There are three difficulty levels: easy, hard, and hardest. And if you enter the password instead of a difficulty level, you may play the game without the problems.

I have only one gripe about the program. When working fractions on the hardest level, there is a 3½ to 5 minute delay before you are told if you are correct. This is apparently due to the time the program takes to work out the solution in fractions rather than in decimal notation. Also, there is a delay while it checks to see if the answer is in lowest terms. Of course, this time could be used by the student to check the problem out for him or her self before the computer gives it all away. But knowing most kids, they'll use the time to go get something to eat!

The program is in BASIC so that you may modify it to fit some of your own needs. However, if you have only 16K of memory, you won't have enough room to modify it much. The program uses all of 16K. This includes the BASIC code, the machine-language subroutine and work area.

The other new program from Med Systems is "The Human Adventure" which allows movement through the human body's cardiovascular system.

Adventurers will find this one a challenge. I haven't accomplished my mission yet. This too, will need all of 16K memory.

Many people will remember a movie called "The Fantastic Voyage." In this film a patient having vital information was incapacitated by a blood clot in the brain. To get to it, the medical team was placed inside a mini-submarine and then reduced to microscopic size. The sub was then placed into the bloodstream of the patient by a hypodermic syringe, enabling the medical team to travel through the patient to the site of the clot and destroy by laser fire.

However, there was one catch: antibodies identifying the submarine as a foreign material began to attach themselves to the sub, signaling the white blood cells to begin their attack.

This is the premises of "The Human Adventure." Therefore, only movement in the direction of blood flow is legal. Access to all major parts of the body is possible by making the correct decision. The CATscan-like graphic of the patient shows the sub's location and the disease locations.

Since this program was designed more as an educational tool than as an Adventure, this program has three modes. The exploration mode involves no disease and no white blood cell attacks. It will allow you to become familiar with the body's layout. All locations are fully described, and you may choose either a female or male body. (May I recommend a dictionary beside your computer. Many medical terms are used. Don't be surprised if your children start talking like surgeons!)

The game mode involves a patient with cancer. Initially there are two affected sites, each growing at an average but not identical, rate. The rate is determined by the difficulty level chosen (easy, moderate, and difficult.) At a certain point in

continued on next page

continued from previous page

their growth they will begin to infect other sites, if they are not destroyed.

Your sub is equipped with a laser gun to kill the attacking blood cells, and you will have to periodically electrify the hull to burn off the build up of antibodies. Of course, all this traveling and laser fire will use energy, so you must make your way to the brain to replenish the energy of your batteries through filled induction.

When you have finally made your way to the affected site, you will have to shoot it with interferon charges. Those cost no energy, but you have only 20 per game.

For reports and other information, the command REPORT will give the locations of all affected sites, the commands available, and the number of interferon charges left. There is also an option for arterial/venous traces.

All commands are one word and may be abbreviated to three letters. You are shown the percentage of energy still left as you travel. If you allow the energy level to drop below critical, you will have no defense and the entire submarine and crew will be engulfed and destroyed by white blood cells!

The third mode is the attention or demonstration mode. In this mode the moves are made by the computer and no disease incurred. Also, after 6 to 7 moves, the venous/arterial traces are shown. Then the demonstration returns to moves.

The only gripe I have about "The Human Adventure" is that the graphics are not as detailed as I would like them. You may mistakenly think you are at the affected site by what the graphics show. You may be in the right phrenic artery and the disease really be in the right phrenic vein, although the graphics picture you right on the site. You have to check out the descriptions with the indicated sites given by REPORT. However, I suspect the graphics are as detailed as can be, given only 16K memory to work with.

I enjoy "The Human Adventure" very much even though I sometimes get very frustrated. I seem to end up going in circles, or

am caught by the current heading in the opposite direction from my intended course! But then, I'm not always going the right way in real life!

Both "The Playful Professor" and "The Human Adventure" are available from Med Systems Software, P.O. Box 2674, Chapel Hill, N.C., 27514. They guarantee all tapes to load and run, and any defective tapes will be replaced, if the bad ones are returned to them. They also guarantee to ship your order within 2 working days. If there is a back-order, you will be notified and the balance of your order sent ahead to avoid delay. The programs are very reasonably priced at \$9.95 each. Not a bad deal, wouldn't you say???



## This Weekend: STIK IT.... ..to your

# TRS-80

That's right! Esmark's VIDJET-STIK light pen has the TRS-80 CONNECTION for LEVEL I & II. Your 4K to 48K TRS-80 System will come alive under your VIDJET-STIK within minutes of its arrival. That's because there are no wires to solder or traces to cut. You're up and running as fast as you can plug the interface into your system's cassette EAR-jack. CLOAD our custom LIGHT-WAVE demonstration software and RUN. And because the interface has a plug for your recorder, you won't have to unplug it again when loading your other software tapes. The interface allows them to pass right thru whenever you're not using the pen. It's exclusive "switched tip" design means the pen's electrically isolated from your system when it's not in use. Just point & press! It's that simple... Plug, CLOAD and RUN. And have we got the software for you to RUN with! Our demonstration tape includes a calibration program (used to adjust the CRT's brightness and contrast) plus STIK-TAC-TOE, AWARI and TOWERS. Two challenging games and a puzzle that will keep grown-ups and children Stiking it to your TRS-80 for hours. And there are instructions provided so you can begin writing your own light pen programs (lightware) for fun or profit (Level II). Or, just sit back and enjoy our LIGHT-WAVE tapes each month. Esmark's unmatched commitment to lightware can bring you up to five new games, puzzles, drills & educational quizzes or simulations each month. Our current LIGHT-WAVE releases are:

- LIGHT-PAK 2: LIGHTPEG (4 peg-jump puzzle!)  
(LEVEL II) ENDRUN (Othello with a twist!)  
LIFE9 (Conway's LIFE with mutations)  
Price: \$19.95 (including postage & handling)
- LIGHT-PAK 3: LITEGAMMON (Backgammon you'll Stik with)  
(LEVEL II) STIKWUMPUS (Caves with a little life)  
MAZEMASTER (Maze after maze to poke thru)  
Price: \$19.95 (including postage & handling)

Order yours now and we'll include a free copy of FLASHBACK, Esmark's newsletter dedicated to the latest news in lightware applications. And don't forget to tell your friends. The VIDJET-STIK can also be ordered for use on most other micro systems using the following processor chips:

8080    Z80    6800    6502

All that's required is a standard cassette jack leading to Ground and a readable single bit input port. Driver software is provided along with instructions for writing lightware applications. And tell your local Dealer that Esmark's got a Dealer package he won't want to miss out on. Delivery is 3 to 6 weeks from receipt of your order. C.O.D.'s are \$3.00 extra but will be shipped within 2 weeks. All prices are F.O.B. Mishawaka, Indiana. Indiana residents add 4% sales tax.

#### ALSO COMING FROM ESMARK

- TRS-80 Printer Interface (Cassette AUX-jack interface for all RS232C prints. Includes LLIST & LPRINT software)
- TRS-80 RS232C Communications Interface (Makes your TRS-80 a full I/O terminal to time sharing systems the world over. Gives you intelligent or dumb terminal capabilities at 110 or 300 BAUD. Also includes Printer Interface above with 20 mA current loop & TTL level I/O options.)

— TRS-80 is a trademark of Tandy Corporation —



**ESMARK** INCORPORATED  
507 1/2 E. MCKINLEY HWY. MISHAWAKA, IN 46544  
(219) 255-3035

\*ELECTRONIC SYSTEMS MARKETING

**\$62.95**  
PLUS \$1.50  
POSTAGE &  
HANDLING

VIDJET • STIK™  
ESMARK INC. MISHAWAKA IN 46544

READER SERVICE 19