Radio Shack

TRS-80 Microcomputer NEWS

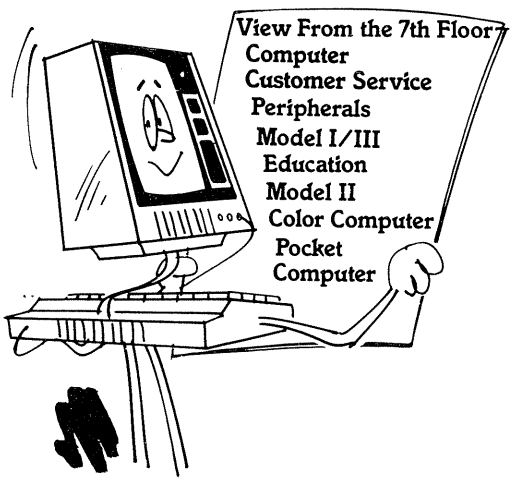
October, 1981

P.O. Box 2910, Fort Worth, Texas 76113-2910

THE MICROCOMPUTER NEWSLETTER PUBLISHED FOR TRS-80 OWNERS

Volume 3, Issue 10

NEWSLETTER INDEX IN THIS ISSUE . . .



The Little Idea That Grew

Eighteen months ago in the March/April, 1980 Newsletter Radio Shack announced a new product—The Applications Sourcebook. The idea was innovative:

- get together a listing of currently available Model I and Model II application software (no system software allowed)
- include short program descriptions, equipment requirements, where the program could be purchased (even if it wasn't Radio Shack!), and what the software would cost
- list the progams in five distinct categories (Business, Education, Specific Industry/Profession, Home Use, and Games)
- and sell the listing in every Radio Shack store in the country for only 99 cents.

At the time we knew that no matter what else happened, we would at least have our products listed. We really had no way of knowing whether we would have 50 listings or 500.

Well, when the first Applications Sourcebook was published it contained over 1000 listings. We were amazed! And they sold like hotcakes. In fact, they sold so fast that we had to go to press with Volume 2 almost before the ink was dry on Volume 1. The second volume contained over 1500 listings, and the third volume, which reached our warehouses in mid-August, contains over 2700 listings. The price has increased a little (to $2.95), the number of computers covered has increased alot (from 2 to 5), and the number of listings has almost tripled.

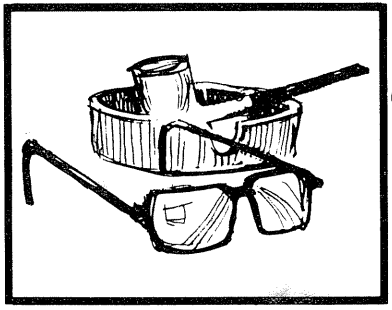
The people responsible for the Application Sourcebook asked us to answer a few of the questions that they get asked quite frequently.

OK, you have a program that you would like listed in the Applications Sourcebook. What now? First, drop by your local Radio Shack and ask for an application. If they don't have one, write to:

TRS-80 APPLICATIONS SOURCEBOOK
RADIO SHACK BOX 17400
FORT WORTH, TEXAS 76102

When you get the application, decide what category your program best fits. If you think your program fits more than one of the categories, and you would like to list it in each of those categories, you must fill out a separate application for each category. There is one other restriction: because of space limitations, you can only list two computers in a single listing. That is, if your program runs on both Model I and III, you can use a single listing.

(Continued on Page 2)



View From the 7th Floor

by Jon Shirley, Vice President Computer Merchandising

This month and last I received some mail and calls protesting our price reduction of the Model III disk system by making it 48K from 32K without a price change. The general theme of these calls and letters was since I just bought one before you reduced the price I should get a refund. If that is logical why should it not be equally logical, if we raise a price, to insist all recent buyers pay us additional money? Price changes are a way of life and lately they have been mostly up and up, not down. We are lucky that technology is helping reduce the price of electronics and computers or at least keeping us even when all other costs are going up.

There is another way to look at this and that is from a philsophic standpoint. It is a Tandy policy to do our best to stay competitive and pass on our cost price reductions whenever possible. For example we are now buying our diskettes in such large quantities that our costs went down. Result? We lowered our sell price by $1.00 on all of them. You can contrast this with the statement made by the president of one of our largest competitors in small computers when their latest quarter showed higher gross profits. "Much of this change," he said, "reflected lower prices for semiconductor components generally . . . . . ." Now what is funny is that this company raised their prices early this year while they were writing the orders for those lower cost semiconductors. We do not think that is a good long term way to do business.

From time to time I have commented on 80 Microcomputing and that drew a letter from the publisher of another TRS-80 only publication, 80-US, wondering why not equal time. Good question and I guess equal time is certainly fair. So is equal reading time so I have been reading 80-US lately. If you have not seen it I strongly recommend you get a copy or subscribe. It is full of timely, well written articles. It has a good balance of articles to advertising and will appeal to Model II users as well as Model I and III users. I hope they can add some Color and Pocket Computer coverage in the future.

The May/June issue (yes it is published bimonthly), contains some neat items including "how to" for random files and PRINT USING (both of which need a lot of how to) and a Model II TRSDOS vs CP/M ™ comparison. If you are interested write to 80-Northwest Publishing Inc., 3838 South Warner St., Tacoma Washington 98409. They also can supply back issues.

Speaking of CP/M, the review of it's use on Model II mentioned how fast it is but did not tell why. One of the reasons is that it assumes the user is an expert and that disk drives never make errors. TRSDOS contains a lot of error checking and recovery and, unfortunately that takes time. For example when you do a Backup, Model II TRSDOS will attempt to fix a directory with an error. In addition, since the directory is the most often read part of the disk and is thus prone to wear, Mod II TRSDOS writes 2 directories. But if you want, and if you will be careful, you can speed up disk activities quite a bit.

First of all you can get rid of the second directory. FORMAT a disk with one directory on it by setting ALT = 00 when you do a FORMAT then do a BACKUP and the second directory will be gone. NOTE: Do not do this with Scripsit. Scripsit formats and backups must be done from Scripsit, it only has one directory anyway and it is not on the normal track.

The next way to speed up disk access is to disable VERIFY DETECT. Verify detect is a function that watches for diskette swaps without notification to the system. Turn this off and it will speed up operations but be sure to do a command I if you swap diskettes. Finally, you can gain speed by turning VERIFY off. VERIFY does a read after write to verify all diskette writes. There is a lot of safety here so turn this off at your own risk. All of the above features of TRSDOS that you can defeat are not in any versions of CP/M that we have seen. And the reason CP/M pops up on the screen so fast is that it does no system diagonstics on power up while TRSDOS does a memory and system test.

By the way I have been doing some writing on the new Model II Scripsit word processing system 2.0. I switched back to 1.0 for this column and now I now I know why I like 2.0 so well . . . it's super fast. If you are a Model II owner and have our original Scripsit or some brand X WP system take a look at Scripsit 2.0 at your nearest Radio Shack that carries Model II's. (And there are now about 700 of them.)In addition to faster operation and a lot of new features there is a dictionary available with over 100,000 words for people who can't spell, like me.

It seems I have been writing about the Model II a lot lately, at least enough to have it mentioned in a couple of letters. That is probably because I have one in my office and I use it a lot with Scripsit, Profile and VisiCalc. But I do have and use our other computers. For example in my living room is a Color Computer which is a beta test site for new game programs. My kids say it's nice to have a dad in the computer biz. Perhaps they will let me play someday. In my den is a Model I disk system, very old but still used quite often. One of it's major tasks is a multi-thousand name charity mailing list that my wife controls. And I never travel without my trusty Pocket Computer. So I got the message, equal time for all TRS-80's. I'll try.

Until next month.

CP/M Trademark of Digital Research

IDEA (From Page 1)

However, if your program runs on Models I, II, and III then you will have to decide which two machines to list in a single listing (obviously you could take two separate listings and cover all three machines.) For each application (one program, one category, two computers only on each) send the application and a check for $10.00 to the address listed above. If you have one program in one category, send one application and ten dollars. If you have one program in two categories, send two applications and twenty dollars. If you have three programs and each one is to be in only one category, you would send three applications and thirty dollars.

What do you get for that $10 per listing? Your listing will be included in every Application Sourcebook published by Radio Shack within the next year, and your year doesn't start until you actually get in a sourcebook. For example, the deadline for the third volume was March 10th, 1981, but we didn't receive your application until March 11th. What now? Well, your information was added to the material which will be published in the fourth volume of the Sourcebook. The one year listing will not begin until the deadline for the fourth volume.

How many Sourcebooks will that be? We really don't know. Those people who listed in the first volume are also listed in the third volume, but won't be listed in the fourth volume unless they re-apply, and again pay $10 per listing. The sourcebook is published "periodically as needed." The printing of each new volume is targeted for quarterly deadlines, and we stock our warehouses based on four volumes a year. Once a volume is finalized, it takes about 90 days to get it printed, shipped and stocked in our warehouses for ordering by our stores.

What happens if I need to make a change to my listing? Revisions to current listings (published or unpublished) should be sent to the above address. Please tell us how the listing appears now (or what information you sent us) and how you want it changed.

COMPUTER Customer Service

Computer Hints, Help and Tips

We have accumulated so many bits and pieces of information this month that we are going to deviate from the usual format of this article (i.e. feature story and most commonly asked questions), and concentrate on getting this information out in an organized fashion. Without further delays let's just dive into the problem areas.

ANNOUNCEMENT

We have recently added another support group to our Computer Customer Service area. This new group will be mainly concerned with the educational software and its applications. If you need help with an educational package, call one of these numbers and ask for the education department :

1-800-433-1679 (outside Texas)
1-800-772-5914 (inside Texas)

ONCE AGAIN

We would like to take this opportunity now to repeat several points that will make information flow more smoothly when you call Computer Customer Services.

We will be able to help you better if you have the following information readily available:

1. Your name, and if applicable your store number.
2. Your phone number, where we can reach you.
3. What TRS-80 computer system you are using and the number of disk drives, if applicable.
4. The name and stock number of the software package you are using. If you know the version number, we would also like that information.
5. What error codes your program is generating.
6. How the error occurred and the function you were excuting when it occurred.
7. Information about any patches or program corrections that you may have made.

Please do not transfer existing software (this includes Applications software as well as BOTH machine language and BASIC programs) onto new operating systems unless specifically instructed to do so.

HANG-UP

Have you ever been in the middle of a program, inputting information from the keyboard and suddenly the machine apparently just stops?For example: you have just received tape mailing list as a birthday gift and are very excited at the prospect of getting that tattered old address book into some semblance of organization. After reading the manual from cover to cover, you begin the task of typing the names and addresses into the computer. Everything is going along quite smoothly until you have typed in around a hundred names and suddenly the machine apparently just stops. In desperation you decide to take a break and forget about the whole thing. After a couple of minutes you return, and to your surprise the cursor is back, flashing and ready to accept information.

The following is a brief explanation of what happened to the computer during those few minutes when it hung up.

For the purpose of this discussion, there are two kinds of string variables (strings):"predefined" and "undefined." A "predefined" string is one which is initialized using an assignment statement, such as A$ = "HELLO." An "undefined" string is a string which is initialized using any of the string functions, such as INPUT A$ or A$ = LEFT$(B$,1). Since the problem does not occur with "predefined" strings (unless you redefine them, in which case they become "undefined"), no further reference will be made to them, and all references to strings will refer to "undefined" strings only.

If you have a string ("undefined") defined as the word "HELLO" (for example, you respond—HELLO to INPUT A$), the string is five bytes long. If you later define the string as "GOODBYE" (you type in "GOODBYE" to the INPUT A$), A$ will be seven bytes long.

This is what happens in memory:

The computer looks at the variable A$. Since it has been redefined, it will take the next seven bytes of unoccupied string space. It will not write over the five bytes of "HELLO," even if the new string length is less than or equal to that of the old. It takes new bytes from string storage, leaving the old bytes as garbage.

After a while, all of the unused string space is taken up by garbage. This is when the program hangs up. The computer is examining the string space to see if there is any garbage it can delete. If there is no more garbage, an out of string space (OS) error occurs. If it finds garbage it can delete, the computer moves all the good strings back to the top of the string space and deletes the garbage at the bottom. This is what takes so much time.

To reduce the time needed for this operation, follow these guidelines:

1. Make all string arrays as small as possible.
2. Minimize string use as much as possible.
3. Clear more string space.

This will not get rid of the delays, but they will occur less frequently and take less time when they do occur.

FREQUENTLY ASKED QUESTIONS

Question 1: I'm using Model III Disk Scripsit with my XYZ serial printer. Will it work?

Answer: We have no way of knowing if all the serial printers on the market will work with Model III Scripsit, but we can at least suggest that you first use SETCOM to initialize the serial port (the parameters for this command will vary, depending on your printer). SETCOM is a TRSDOS command. To see the proper syntax for this command, either type in HELP SETCOM at TRSDOS READY or refer to the Disk Operating Manual for Model III. After you have initialized the serial port via SETCOM, go into SCRIPSIT and print using the sequence **[BREAK][P][,][S]** command sequence.

Question 2: I have converted my Model I Accounts Receivable System (ARS) or Accounts Payable System (APS) program onto a Model III disk and now I can not change the date.

Answer: If you can not change the date it is usually because you have not converted to the newest version of these programs. Please make sure you are using the 3.0 version of ARS or APS.If you do not have version 3.0, have your local store order stock number 700-2002 for Accounts Receivable version 3.0 or 700-2001 for the 3.0 version of Accounts Payable.

Question 3: I'm getting an error code 58 or a disk I/O error in my MEDICAL OFFICE PACKAGE on Model III when doing the system initialization or when trying to set up my patient disk.

Answer: There are two probable causes—

1. Your data disks are not formatted.
2. Your disks are not in the proper disk drives—you need to make sure the patient data disk is in drive number three (the drives are numbered 0,1,2,3). (Continued on Page 4)

COMPUTER SERVICES (From Page 3)

Question 4: I'm playing Raaka-Tu™ but I can only get 20 of the possible 25 points. Why?

Answer: Hint, there is a magic gem worth 5 points, which appears in a different room each time you play the game. When you are searching for this gem it is not necessary to carry any extra supplies.

Question 5: Is there anyway to printout the formulas from a VisiCalc sheet?

Answer: Yes. The sequence of commands to use is **[/][S][S][:P]**

Question 6: How do I use the Color Computer commands CSAVEM, CLOADM, and DLOADM?

Answer: Here is the correct color computer syntax for the machine language input/output statements: CSAVEM and CLOADM on color computer:

CSAVEM "filename," addr1, addr2, addr3
- filename—a filename you specify of 8 characters or less.
- addr1—decimal start address of the machine language routine
- addr2—decimal end address
- addr3—decimal transfer or execution entry point address

CLOADM "filename," offset
- filename—the filename of machine language routine to be loaded
- offset—decimal offset for the start address (optional)

DLOADM—we do not have the information necessary to support this feature at this time.

Question 7: Can you give me some help in hooking a serial printer to my TRS-80 Model I, II, or III?

Answer: Here are some hints on connecting a Radio Shack computer with an RS-232-C serial printer.

COMPUTER PIN #	RS-232-C SERIAL PRINTER PIN #
1	1
2	3
3	2
7	7

4) THESE
5) PINS
6) ARE
8) TIED
20) TOGETHER

*Note: Model I/III users must have an RS-232 card in their units. The Model II comes with RS-232 already installed.

Model I users should run the DECWRITER program in their RS-232 manual.

Model II users should do a SETCOM then a FORMS S to indicate the use of a serial printer.

Model III users will need to use the serial printer option on Model III Scripsit, wait until we come out with a program for them, or write their own machine language serial driver program (all necessary information is in the owner's manual for machine language programmers).

Next the number of bits per word, parity, and stop bits need to be set as per your printer manual. The baud rate setting with the above wiring diagram should not be set any higher than the speed at which the printer prints (300 is average).

Question 8: What general steps should I follow when I get an error in one of my Model II business software packages?

Answer: On Model II business packages, any abnormal exit from the program will generally cause an error in the data file. Therefore if at any time, for any reason, you receive an error, or the program goes back to TRSDOS without an error (but you didn't tell it to), the ONLY safe thing to do is to immediately go to yesterday's BACKUP, make a BACKUP of it, and redo today's information.

If the program giving the error is re-run, it is possible for it to continue without another error until the end-of-period, when the program accesses the entire data file, and by this time the error will be on all your current backups. Unless you are also keeping monthly backups, your only course of action will be to begin again from the initial setup.

Question 9: I am using PROFILE II, and I need to sort information by date, including the year. How can I do this?

Answer: If you want to sort a date (such as August 4, 1981) numerically, you can either use a numeric field instead of alphanumeric and use the form 810804 instead of 08/04/81, or use alphanumeric fields and either enter the information as 81/08/04 or set the month and day as one field and the year as a separate field.

COMPUTER CUSTOMER SERVICES ADDRESS AND PHONE NUMBERS

8AM to 5PM Central Time
Computer Customer Services
400 Atrium, One Tandy Center
Fort Worth, Texas 76102

Model I/III Business Software
- Outside Texas 1-800-433-5641
- In Texas 1-800-772-5973

Model II Business Software
- Outside Texas 1-800-433-5640
- In Texas 1-800-772-5972

Education Software
- Outside Texas 1-800-433-1679
- In Texas 1-800-772-5914

All Other Calls Related to Computers
- Outside Texas 1-800-433-1679
- In Texas 1-800-772-5914

Switchboard - 1-817-390-3583

Subroutines

Robert Crockett Blacksburg, Virginia

As a Research Assistant for the Productivity Evaluation Center at Virginia Polytechnical Institute & State University, part of my job is developing software for the Model I Level II TRS-80 microcomputer. Below are two subroutines which you may find useful.

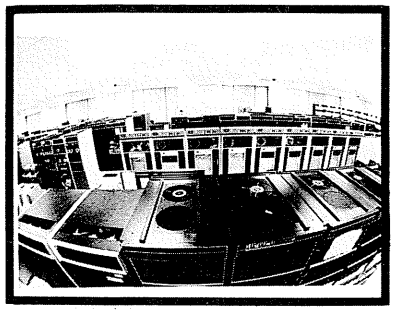
The first subroutine is called "LPCHK/BAS". This routine simply checks to see if the line printer is turned on. If the line printer is on then a RETURN is executed, otherwise the terminal screen is cleared and a message is printed advising the operator to turn the line printer on.

LPCHK/BAS

```
60000 REM BOB CROCKETT APRIL, 1980
60010 IF PEEK(14312)<> 255 THEN RETURN
60020 CLS
60030 PRINT
    : PRINT
    : PRINT
    : PRINT
    : PRINT
    : PRINT
60040 PRINT CHR$(23)
60050 PRINT TAB(5); "TURN LINE PRINTER ON!"
60060 IF PEEK(14312)<>255 THEN CLS
    : RETURN
60070 GOTO 60060
```

Note: this routine only determines whether or not a line printer is attached and turned on. The printer still may not be available for printouts.

(Continued on Page 26)



CompuServe

CompuServe Information Service

Personal Computers Can Make Banking—and Life—Easier

There are people whose job it is to spend eight hours a day, five days a week, just thinking about the future. They're called "futurists." They address themselves to such weighty questions as: What technologies will be in existence 20, 30 or 50 years from now? What problems will our civilization face? Which resources will be scarce, which plentiful?

The futurists also focus on life-styles. Picture father, mother and 1.7 children ordering dinner from Max, the family robot, who also serves as housekeeper and guest greeter. Picture traveling in motorized vehicles that whisk us to an office-in-the-sky in something less than 30 seconds. Better yet, picture not going to the office at all, but conducting all your work via your handy home computer. You may never leave your Basic Life-Sustaining Structure (i.e.: house) for weeks at a time. Instead, mere button-pushing is adequate to perform the myriad of functions that comprise living in the futuristic society.

While the futurists eagerly delve to discover what lies ahead, the rest of us are filled with . . . well, trepidation. Fear, even. And perhaps a healthy dose of curiosity.

The negative emotions, like fear and trepidation, tend to overshadow the positive aspects. Our life-styles, we argue, are just fine as they are, thank you. Push-button this and computerized that are innovations this generation can do without, we say. I'll do my shopping in the grocery store, my banking at the nearest branch office. It's just more human, we say, than pushing buttons.

Perhaps. There IS something to be said for the grocery checkout clerk's smile and for the bank teller's parting, "Have a nice day!"

On the other hand, there is a great deal more to be said for convenience. Consider, for a moment, the strides we have made during just the past several decades, strides that have been greeted with cheers and thanks and such remarks as, "How did we ever get along without it?"

Banking is a good example. Your bank, for instance, most likely came into being with one downtown main office. Its customers went to that downtown office to transact any business, either professional or personal. Then came the movement to suburbia, as people fled cities to embrace the advantages of split-level homes with two-car garages and quarter acres of grass to mow. The banks promptly followed, establishing branch offices on major suburban streets. Your money was suddenly as close as the nearest streetcorner.

Still more changes were in the offing, all in quest of what had become a major watchword: Convenience. Suburban Americans came to rely increasingly on the automobile for transportation. The bank's answer? Drive-in windows, offering customers the opportunity to conduct their banking business without climbing out of their cars.

As society became increasingly mobile, another convenience was introduced: 24-hour banking. "Bankers' hours" were no longer sufficient to serve customers' diverse needs and so automatic tellers were installed, complete with code numbers and a set of instructions for easy operation.

The '70s were a time of another kind of change, as people began the task of coping with dizzying demands on their time and also began worrying—more than absently—about the cost of gasoline. All of this running to-and-fro, to pick up a suit at the dry cleaner's, to buy a couple of steaks at the butcher's, to cash a check at the bank, was beginning to take on real economic significance as gas prices escalated. Wasn't there a way to get these jobs accomplished without all that driving? Indeed, there was, from a banking standpoint at least. The financial industry instituted bill-paying services that could be conducted from the customer's own home telephone.

All of these steps meant progress. They're viewed now as necessary, even vital, steps in the evolution of today's financial institution industry. They have resulted in improved convenience from a consumer point of view. If any thing, these steps—from branch offices to 24-hour service—have made banking MORE human, for they have resulted in very substantial benefits that consumers would never have dreamed of in earlier times. Far from dehumanizing our society, these conveniences give us the freedom to enjoy our leisure hours to a degree only dimly imagined a half-century ago.

The latest of these conveniences, called Bank-at-Home, promises even more advantages to today's consumer. It's what the futurists have been promising us for years. A new world and a wealth of information is at our finger tips. And what can be more "human" than that?

"Someday" is Now with CompuServe, Radio Shack and United American Bank

Someday our regular visits to the neighborhood bank will become less regular, not because we have less money, but because we will all have a more efficient way to conduct our banking transactions. For residents of Knoxville and Memphis, Tenn., "someday" is today.

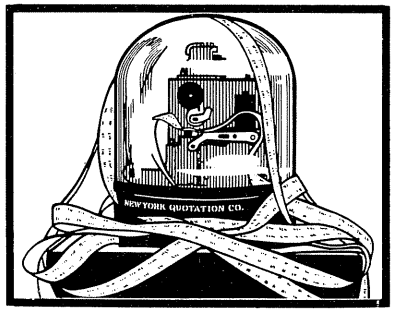
Electronic home banking is alive and well and working for customers of United American Bank in those cities. (You can access Home Banking through main menu item 2, Finance).

CompuServe, Radio Shack and United American Service Corporation have joined forces to introduce the first commercially available home banking service in the country. United American Bank is the first bank to offer the service, called Bank-at-Home, to its customers.

Products that the customer is already familiar with—a typewriter-like keyboard, a telephone and a television set—enable him or her to perform a variety of banking transactions without ever leaving home, say bank officials. The Bank-at-Home service potentially allows United American Bank customers to pay most of their bills, receive current information on their checking accounts, develop a computerized bookkeeping and tax record service, apply for loans, get national, international, financial and stock information and communicate messages from one computer user to another. The initial group of 400 customers pay monthly for the service, which can include the use of Radio Shack's TRS-80 Color Computer.

The system, as the first of its kind, takes some getting used to on the part of consumers, and United American Bank takes that fact into consideration. Explains Thomas E. Sudman, UASC president, "We're releasing these programmed services in phases to allow our customers adequate opportunities to familiarize themselves with in-home computer use." He adds, "As their expertise and needs increase, the sophistication of the information services increases."

(Continued on Page 26)



Dow Jones

Dow Jones Information Services

Dow Jones Information Services will be an ongoing contributor to the TRS-80 Microcomputer News.In this first article we would like to share with you a personal story we received from a new subscriber.

"HOW I LEARNED TO LOVE MY COMPUTER WITH DOW JONES INFORMATION SERVICES."

My husband Phil walked through the front door a few days ago so excited he could hardly speak.

"Look Madge, we have a new member of the family."I began to share his excitement. I've always loved animals and thought we had a new puppy to play with. My thoughts were quickly interrupted.

"Help be bring the boxes in from the car."

Boxes? I thought.

This was the beginning of our family relationship with our brand new TRS-80 Model III computer.

"I'd rather have a dog" I mumbled.

"Did you say something dear?" Phil asked.

"Yes, please open the door."

That evening Phil spent hours setting up his computer and trying out his new software. I learned that software meant computer programs, not sweaters. I retired for the evening listening to pecks and whirs from downstairs.

The next day, as Phil left for the office, he reminded me to be careful with "George," and that maybe I shouldn't even go near him.

George? Already he had a name for the computer. It had taken us two weeks to name our son. And now, with the house empty, I felt the presence of this new intruder. With some hesitation I decided to look the situation over. I was feeling a bit lonely and thought perhaps George could talk as well as balance checkbooks. I looked over the machines and noteboks containing the software. At last! A familiar name. There was a booklet called the Dow Jones Information Services User's Guide. I know that Dow Jones publishes the Wall Street Journal and Barrons and provides stock quotes and averages. But what was it doing in with all this computer equipment?

I was now intrigued to the point of learning this system and began reading to find out how to turn George on. After mastering disk drives and modems, I picked up the material on Dow Jones. Apparently it was part of the Radio Shack Videotex program. In addition to the User's Guide there was a user's agreement that said, "telephone us for your unique personalized user password." This was terrific—I could talk to Dow Jones. I found the toll-free number on the contract which I quickly memorized: 1-800-257-5114 or in New Jersy, 609-452-1511, and then gave them a call.

"Hi, I'm Madge Midway and I'd like a password for my Videotex package."

After a few easy questions I had my free password and was told that I would receive more information on using the system and pricing information in the mail. I was also given a local Tymnet phone number to use when calling the service. So far so good. Why even at 8:00 a.m. there was someone there to answer the phone, and the Service is available 21 hours a day. And I have an hour of free time during non-prime hours to explore this new electronic service.

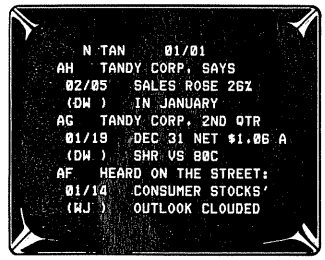
However, I couldn't wait for that evening so I opened my User's Guide and saw that I could get news from The Wall Street Journal and Barrons, and the Dow Jones News Wire, as far back as 90 days, and as recent as 90 seconds. "That's exciting!" I thought and I flashed on part of a news story I had heard on the radio that morning about a new Sears catalog. Maybe I could get the whole story from Dow Jones Information Services (DJIS). Bravely I faced the steps to start up the service. As I read through each step, I realized I didn't know the symbol for Sears. It would be arriving by mail in the Dow Jones Operating Guide and Directory. Not to be deterred, I called the DJ Hotline and they graciously told me the symbol for Sears as well as a few other companies I was interested in. And so I proceeded with the step-by-step directions to "log on" (another new term meaning to hook my computer to DJIS). It was amazing that the computer could ask questions and interpret my answers, and in just a few moments I was communicating with Dow Jones.

According to the User's Guide I could get the most recent story, headlines, or quotes on companies or industries. I carefully followed the directions and examples given in the User's Guide:

To Get The Headlines On A Company

Desired Information	Procedure	Example
To scan the most recent headlines on a company	1. Type . (period) 2. Type a company code 3. Press **[SPACEBAR]** 4. Type the number [01] 5. Press **[ENTER]**	[.TAN 01]**[ENTER]**

Your screen should look like this:



To get to the next page and more headlines, press **[ENTER]** again.

Then I tried the same procedure for Sears: .S 01 . In a flash all this information was printing on my screen about Sears. And there was the story on the new Sears catalog. I pressed the 2-letter code indicating the story I wanted, and settled comfortably in my chair to read the article.

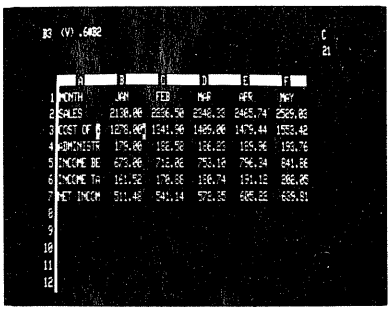
"This is so easy," I said to George. "I think I'll catch up with all the news I'm interested in."

The listing of codes for different subjects was in the back of the guide so I looked up General News (I/GEN), Executive Branch News (G/EXE) to see what the President and Congress were up to, Taxes (TAXES) so I could impress my tax attorney husband with my knowledge of recent tax legislation, and much more, selecting only those stories I wanted by first viewing the headlines.

I have really never followed the stock market very carefully, but Dow Jones Information Services offers the latest quotes (delayed only 15-30 minutes from actual prices on the floor of the exchange), not only on common stocks, but also on warrants, bonds, Treasury issues, mutual funds, and options. Composite quotes are available from 6 different exchanges, or individual quotes from the New York, American, Midwest and Pacific exchanges, plus the Nasdaq Over-The-Counter market.

I checked the price on a stock we bought recently and discovered it had jumped 2½ points just this morning. Quickly I retrieved the latest story for the company and saw that a new oil field had been discovered.

(Continued on Page 25)



VISICALC™

The Visible Calculator

VisiCalc Sort

Alexander B. Spencer Dallas, Texas

I recently found it convenient to sort a VISICALC table. The enclosed program was devised to accomplish that end using a 64K TRS-80 Model II.

In short, that portion of the VISICALC table to be sorted is saved as a DIF file using the /S# command and the enclosed program is used to read, sort and save the sorted file in DIF format. The sorted file is then loaded under VISICALC using the /S#L command.

If only a part of the VISICALC table is to be sorted only that part is saved. After sorting it is loaded after reloading the original file. This technique is possible when loading VISICALC data from a DIF file.

The program consists of a BASIC driver based on DIF read and write as given in the appendix of the VISICALC manual and the n-dimensional assembly language sort recently published in the NEWSLETTER.

A DO file can be called from VISICALC using /SE "DO xxxx" that will load BASIC, load the read-write-sort program, and then return to VISICALC where the sorted file may be reloaded.

Cues should be self-explanatory and should cause the user little trouble.

Listing 1.

Here is Mr. Spencer's Program:

```
10 REM      *****PROGRAM DIFSRT/BAS*****
20 REM
30 REM      * REQUIRES 'SORTN'
40 REM      * MEMORY ABOVE 60500 PROTECTED
50 REM      * ABS - MAY 80 (VERSION 1.0)
60 REM
70 REM - This program reads a DIF file and sorts it by rows
     or columns,
80 REM - then writes it to another (or the same) file in
     DIF.
90 REM
100 CLS
110 CLEAR 5000, 60500
120 DIM T(50)      ' Maximum of 50 vectors or tuples
130 DIM V(50)      ' T, V and V$ hold the
140 DIM V$(50)     ' Type - Indicator, Number
150 DIM A$(50,50) ' Array to be sorted
160 DIM Z(4)
170 REM ***** ROUTINE 'DIFRD' *****
180 REM - This routine reads a DIF file into memory in
     string form.
190 REM
200 GOSUB 370
210 GOSUB 440
220 FOR I=1 TO NT
230 IM=I-1
240 GOSUB 640
250 FOR J=1 TO NV
260 JM=J-1
270 IF T(J)<>0 THEN 300
280 A$(JM,IM)=STR$(V(J))+"0"
290 GOTO 320
300 IF T(J)<>1 THEN 320
310 A$(JM,IM)=V$(J)+"1"
320 NEXT J
330 NEXT I
340 GOSUB 840
350 CLOSE 1
360 GOTO 950
370 PRINT "FILE NAME";
380 INPUT F$
390 PRINT
400 OPEN "I", 1, F$
410 NV=0
420 NT=0
430 RETURN
440 INPUT#1, T$
450 INPUT#1, S, N
460 INPUT#1, S$
470 IF T$="VECTORS" THEN 510
480 IF T$="TUPLES" THEN 570
490 IF T$="DATA" THEN RETURN
500 GOTO 440
510 NV=N
520 IF NV<=50 THEN 440
530 PRINT NV;" VECTORS. THIS PROGRAM HANDLES ONLY 50."
540 PRINT
550 CLOSE 1
560 STOP
570 NT=N
580 IF NT<=50 THEN 630
590 PRINT NT;" TUPLES.  THIS PROGRAM HANDLES ONLY 50."
600 PRINT
610 CLOSE 1
620 STOP
630 GOTO 440
640 GOSUB 750
650 IF T1<>-1 THEN 780
660 IF S$<>"BOT" THEN 780
670 FOR K=1 TO NV
680 GOSUB 750
690 IF T1=-1 THEN 780
700 V(K)=V1
710 V$(K)=S$
720 T(K)=T1
730 NEXT K
740 RETURN
750 INPUT#1, T1, V1
760 INPUT#1, S$
770 RETURN
780 PRINT "ERROR IN FILE FORMAT."
790 PRINT
800 CLOSE 1
     : STOP
810 REM ***** VIDEO LIST ROUTINE *****
820 REM - This routine lists file (from memory) to
     video display.
830 REM
840 FOR J=0 TO NV-1
850 FOR I=0 TO NT-1
860 PRINT TAB(I*10) LEFT$(A$(J,I), LEN(A$(J,I))-1);
870 NEXT I
880 PRINT
890 NEXT J
     : PRINT
     : PRINT
900 RETURN
910 REM ***** SORT ROUTINE *****
920 REM - Routine to sort 'A$' on any selected col or
    row into
930 REM - alphabetic descending order.
940 REM
950 SYSTEM "LOAD SORTN"
```

(Continued on Page 8)

VisiCalc (From Page 7)

```
96Ø DEFUSRØ=&HEDØØ
97Ø DEFINT Z
98Ø Z=Ø
99Ø LINE INPUT "SORT ON COL OR ROW, OR QUIT (C/R/Q)?
   ";Z$
1ØØØ IF Z$="Q" THEN 172Ø
1Ø1Ø IF Z$="C" THEN 1Ø6Ø
1Ø2Ø IF Z$<>"R" THEN 99Ø
1Ø3Ø GOSUB 12ØØ
1Ø4Ø PRINT "SORT ON WHICH ROW (1 TO ";NT;")";
     : INPUT ZC
     : ZC=ZC-1
1Ø5Ø GOTO 1Ø7Ø
1Ø6Ø PRINT "SORT ON WHICH COLUMN (1 TO ";NT;")";
     : INPUT ZC
     : ZC=ZC-1
1Ø7Ø PRINT
1Ø8Ø Z(Ø)=NV
1Ø9Ø Z(1)=VARPTR(A$(Ø,ZC))
11ØØ Z(2)=NT
111Ø IF VARPTR(A$(Ø,ZC))<Ø AND VARPTR(A$(Ø,Ø))>Ø
   THEN 112Ø ELSE 113Ø
112Ø Z(3)=65536+VARPTR(A$(Ø,ZC))-VARPTR(A$(Ø,Ø))
     : GOTO 114Ø
113Ø Z(3)=ABS(VARPTR(A$(Ø,ZC))-VARPTR(A$(Ø,Ø)))
114Ø IF VARPTR(A$(Ø,1))<Ø AND VARPTR(A$(Ø,Ø))>Ø
   THEN 115Ø ELSE 116Ø
115Ø Z(4)=65536+VARPTR(A$(Ø,1))-VARPTR(A$(Ø,Ø))
     : GOTO 117Ø
116Ø Z(4)=ABS(VARPTR(A$(Ø,1))-VARPTR(A$(Ø,Ø)))
117Ø Z=USRØ(VARPTR(Z(Ø)))
118Ø IF Z$="R" THEN GOSUB 12ØØ
119Ø GOSUB 84Ø
     : GOTO 129Ø
12ØØ IF NT>NV THEN X=NT
     : GOTO 122Ø
121Ø X=NV
122Ø FOR I=Ø TO X-1
123Ø FOR J=I TO X-1
124Ø SWAP A$(I,J),A$(J,I)
125Ø NEXT J
     :NEXT I
126Ø SWAP NT,NV
127Ø RETURN
128Ø REM ***** ROUTINE 'DIFWRT' *****
129Ø REM - This routine creates a Data Interchange
   Format file.
13ØØ REM - Data may be either numeric (type Ø) or
   string (type 1).
131Ø REM
132Ø PRINT "FILE NAME";
133Ø INPUT F1$
134Ø IF F1$<>"" THEN F$=F1$
     : GOTO 135Ø
135Ø PRINT
136Ø OPEN "O", 1, F$
137Ø GOSUB 158Ø
138Ø FOR I=1 TO NT
139Ø IM=I-1
14ØØ T=-1
     : V=Ø
     : S$="BOT"
141Ø GOSUB 165Ø
142Ø FOR J=1 TO NV
143Ø JM=J-1
144Ø T=VAL(RIGHT$(A$(JM,IM),1))
145Ø V=Ø
     : S$="V"
146Ø A$(JM,IM)=LEFT$(A$(JM,IM), LEN(A$(JM,IM))-1)
147Ø IF T=Ø THEN V=VAL(A$(JM,IM))
     : GOTO 149Ø
148Ø IF T=1 THEN S$=A$(JM,IM)
149Ø GOSUB 165Ø
15ØØ NEXT J
151Ø NEXT I
152Ø T=-1
     : V=Ø
     : S$="EOD"
153Ø GOSUB 165Ø
154Ø CLOSE 1
155Ø PRINT "FINISHED CREATING DIF FILE ";F$
156Ø PRINT
157Ø GOTO 172Ø
158Ø PRINT#1,"TABLE"
     : PRINT#1,"Ø,1"
     : GOSUB 163Ø
159Ø PRINT#1,"TUPLES"
     : PRINT#1,"Ø";NT
     : GOSUB 163Ø
16ØØ PRINT#1,"VECTORS"
     : PRINT#1,"Ø";NV
     : GOSUB 163Ø
161Ø PRINT#1,"DATA"
     : PRINT#1,"Ø,Ø"
     : GOSUB 163Ø
162Ø RETURN
163Ø PRINT#1,CHR$(34);CHR$(34)   'See appendix on
   quoted
164Ø RETURN  'strings in BASIC
165Ø PRINT#1, T; ","; V
166Ø IF S$="V" AND T=Ø THEN 17ØØ
167Ø IF S$="BOT" OR S$="EOD" THEN 17ØØ
168Ø PRINT#1, CHR$(34); S$; CHR$(34)
169Ø RETURN
17ØØ PRINT#1, S$
171Ø RETURN
172Ø PRINT TAB(38)"FINIS"
     : FOR L=1 TO 1Ø
     : PRINT
173Ø SYSTEM
174Ø END
```

Editors Note: We found this program to work quite well, but we do have several comments. First, set the file name and DEFUSR in lines 950 and 960 to the values that fit the way you have the sort routine from the March, 1981 Newsletter saved. We saved it as "SORT/CIM" and ours starts at hex F121 rather than hex ED00 for Mr. Spencer. Our second comment is that we were not happy with the results from a sort of numeric information. Try Mr. Spencer's way, then change line 280 to read:

```
28Ø A$(JM,IM)=RIGHT$(STRING$(2Ø,32)+STR$(V(J))+"Ø", 2Ø)
```

This will right justify the numeric values with blanks to the left. By doing this, numbers of different lengths sort correctly. No further conversion seems to be needed, when VisiCalc reads the numeric values it chops the leading blanks and handles the values properly.

Our next comment is to be sure and include only one data type in your DIF file.

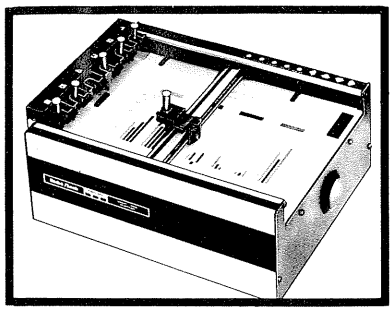
For those of you who may have missed the n-dimensional sort routine in March:

Listing 2.

Here is a Hex listing of the sort program if you are using DEBUG to enter the machine code. Note that the first four digits are the Hex beginning address for that line of entries:

```
FØØØ   ØØ ØØ ØØ 5E 23 56 ED 53   2B FØ 23 5E 23 56 ED 53
FØ1Ø   19 F1 23 5E 23 56 ED 53   1B F1 23 5E 23 56 ED 53
FØ2Ø   1D F1 23 5E 23 56 ED 53   1F F1 21 ØØ ØØ 22 17 F1
FØ3Ø   ED 5B 17 F1 CB 3B AF CB   3A 3Ø Ø2 CB FB ED 53 17
FØ4Ø   F1 7A B3 C8 2A 2B FØ ED   52 22 13 F1 21 ØØ ØØ 22
FØ5Ø   11 F1 2A 11 F1 22 ØF F1   2A ØF F1 ED 5B 17 F1 19
FØ6Ø   22 15 F1 EB 21 ØØ ØØ 19   19 19 E5 ED 5B ØF F1 21
FØ7Ø   ØØ ØØ 19 19 19 ED 4B 19   F1 Ø9 EB E1 Ø9 E5 D5 ØE
```

(Continued on Page 26)



Peripherals

Product Line Manager's News

Last month I promised you more goodies this month and I'm pleased as punch to be able to keep that promise. My subject this time is GRAPHICS. Those of you who have been in to your Radio Shack store or Computer Center to pick up the new Fall catalog, 341, already have an idea of what I am talking about. If you have seen the new computer catalog, RSC-6, you know that we REALLY mean business when it comes to graphics. I'm speaking of the new Digitizer (26-1195-$449.00) and the multi-pen plotter (26-1191-$1995)!

These low-cost, state-of-the-art products bring exciting new capabilities to all our full size systems. The first product I want to describe is the new Digitizer. What IS a digitizer, you ask? This device will allow you to input, directly to your computer, coordinate data contained in maps, charts, photographs, and even your CRT screen. I can guarantee that you will NOT find a device of its capabilities at anywhere near its price. It can accurately define 100 points per inch. We have been able to input four discrete coordinates for the corners of the period printed by a dot matrix printer. (I don't know why you would want to do such a thing, but you can do it with the Radio Shack digitizer.)

This little marvel consists of a base platform containing a carrier which moves in the "Y" axis. This carrier also holds a cross member (arm) which moves in the "X" axis. At the end of this arm is a lucite "cursor" marked with cross hairs. As the cursor is moved about in the active field (11" X 17"), optical counters keep track of the relative "X" and "Y" coordinates of the cursor. The numbers corresponding to the current coordinates are sent to the computer in ASCII form (XXXX YYYY CR) either in a continous stream or only when you press a button on the cursor.

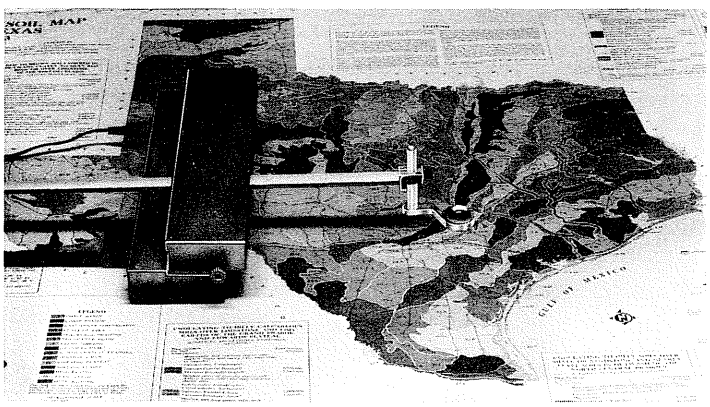


Photo 1. Digitizer

There is no expensive analog to digital conversion needed. The information is generated in digital form and converted, by a built-in microprocessor, to ASCII digits. The information is then sent, via a standard, simple RS-232C interface, to your computer. The result is an accurate, low-cost instrument with many unique features. This device can digitize from almost anything. A very large chart or drawing can be "digitized" by dividing the area to be digitized into 11" x 17" sections, and then simply laying the entire unit on the surface to be digitized and moving from section to section. Software supplied with the unit will adjust for "skew" automatically so that coordinates are always in a completely rectangular format.

The mobility of the cursor in its tracks will even allow you to input digital information from the pages of a book or other fairly thick object. You could easily use this unit on a light table for reading information from transparencies. And last, but not least, the active unit swivels from its base to allow you to place the unit on top of a CRT: The cursor then folds down in front of the screen, allowing you to input data directly from the screen.

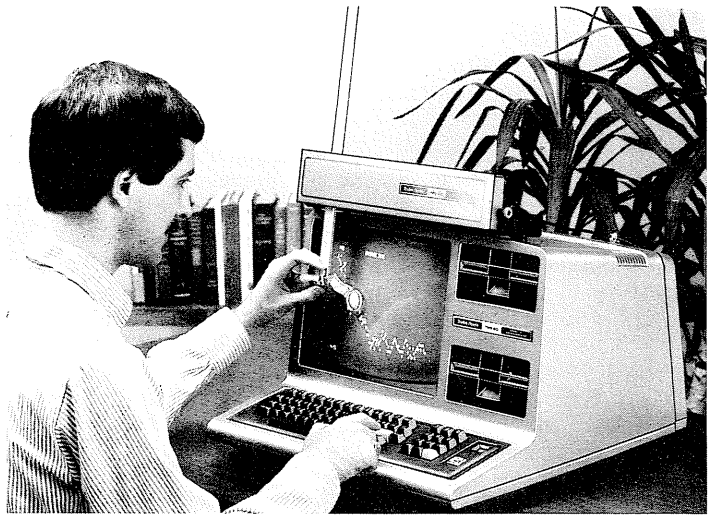


Photo 2. Digitizer used with video screen

If you are using the unit with TRS-80 Models I, II, or III, then in addition to the skew adjustment already mentioned, software is supplied which can compute line length and area. The resulting data is presented in a form that will allow generation of tables or files of information for use by more ambitious programs.

All in all, I believe that the Digitizer is a valuable tool, one that greatly expands the horizons for business as well as scientific purposes. I know of one user who plans to use the digitizer to input the data contained in a number of widely available stock market charts and graphs to his stock analysis programs. Engineering applications are numerous. The ablilty to input data directly to the computer (even the color computer) and then use that data to draw directly on the screen is really exciting.

From this point in time (Mid-August) it looks like we will start filling orders in November. The price looks low in comparison to other units available, but believe me Radio Shack's Digitizer is an accurate, useful tool.

(Continued on Page 10)

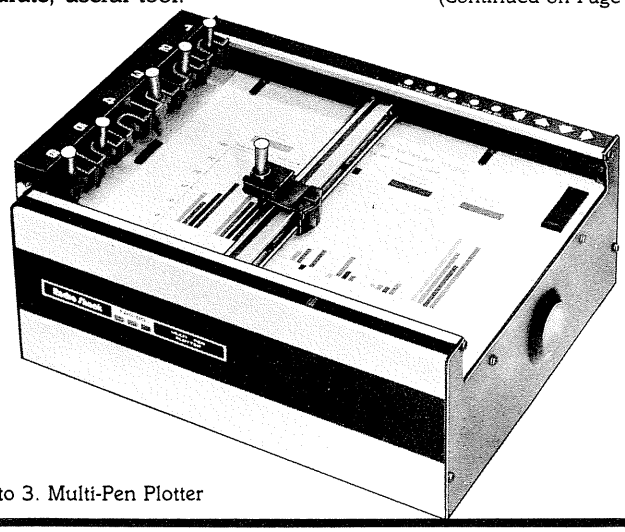
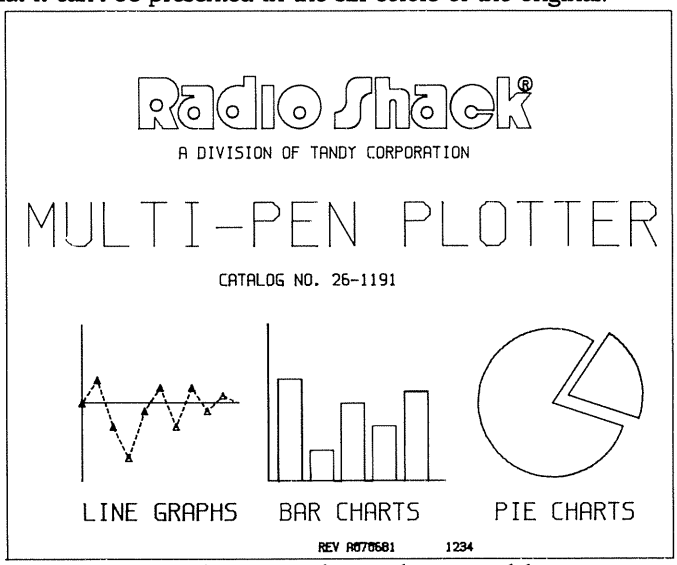


Photo 3. Multi-Pen Plotter

Peripherals (From Page 9)

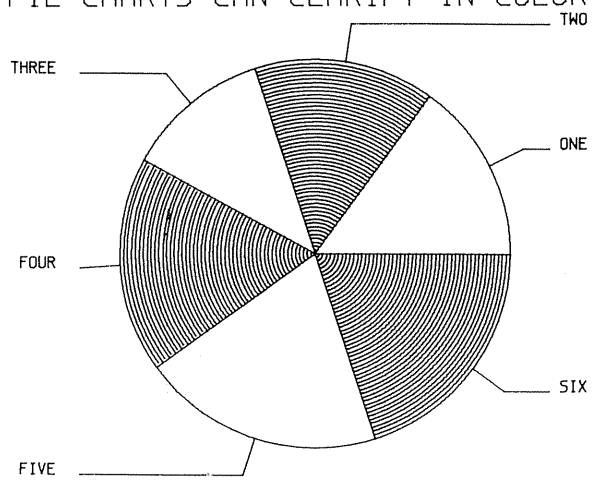
Those of you who have seen RSC-6 should have noticed the new multi-pen plotter (requires RS-232C) added this year. Multi-pen means multi-color of course. This flat-bed plotter accepts 8½" by 11" paper or plastic film. A "stable" of six pens is attached to the side of the plotter. Upon receipt of simple ASCII commands the plotter will select the pen with the color of your choice, draw your graph or chart (or the portions you specify), and replace the pen when finished, and grab another—all untouched by human hands!

The plotter accepts either absolute or relative coordinates given as ASCII digits and draws a variety of line styles (solid, dotted, dashed, etc.), as well as arcs and circles, on command. In addition, alpha-numeric characters and other symbols can be drawn in nine sizes and four rotations. Included here is a black and white copy of the output obtained in the self-test mode. I'm sorry that it can't be presented in the six colors of the original.



The commands are similar to those used by our current single pen drum plotter. They are not the same, however. Each device has its own characteristics relative to plotting field, etc. Any serious application will have to either be dedicated to a specific device or graphing programs will need to be written in a "device independent" format and any particular hard copy machine supported by an appropriate driver. It is my feeling that Radio Shack will, at times, follow both of these formats.

As simple as our "intelligent" plotters are to use, its quite a chore to generate very complicated graphs (other than mathematical functions) on these machines. The software supplied with the multi-pen plotter takes a giant step toward making it easy for you. A disk-based program is supplied with the plotter (in mid-August we are not sure what, if any, media will be supplied. Probably a cassette for Models I and III, possibly (no promises) a disk for Model II. In any event, a printed copy of the source code will be supplied.) which will generate, save, and recall pie charts, bar charts and line graphs from a simple question and answer "menu" format. By simply supplying information regarding titles, scales, legends, and colors, complete graphs can be formatted, ready to accept the actual data needed to complete the picture, from the keyboard. (The chart used to illustrate the plotter in RSC-6 was generated with this software.)



You will easily be able to generate attractive, information-packed, business related graphs with the program as supplied.

I almost forgot to mention that the new plotter includes a set of touch sensitive membrane switches that allow you to manually control pen motion. I did mention that the plotter accepts clear plastic film for generating transparancies. These can be useful for overhead projection or photographic purposes. Please note that the special pens required for this use are listed in the catalog in addition to the ball point pens for drawing on paper. We will also offer the plastic film. Don't use the ball points on film or vice-versa; each pen does its own thing!

The multi-pen plotter and its accessories should be available by November. If you're "into graphics" get one of these machines. Don't forget that our single pen plotter is still available. It is an extremely accurate and dependable device at a much lower price (26-1190A $1460.00 plus cable) for those on a more strict budget. Pens of various colors which fit the single pen plotter are widely available around the country. If you don't mind changing pens by hand, you can obtain multi-color charts with this device also.

That is all I can cram into this month's space, but I promise again, more to follow next month.

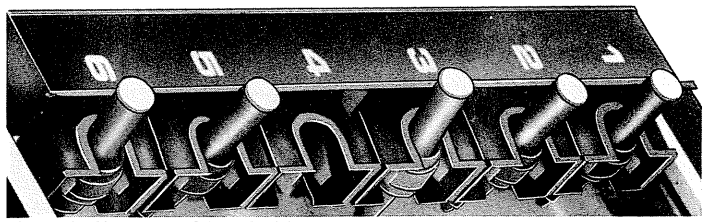


Photo 4. Close-Up of Pen Stable

Here is a summary of the specifications for the Multi-Pen Plotter (26-1191):

Physical Dimensions:

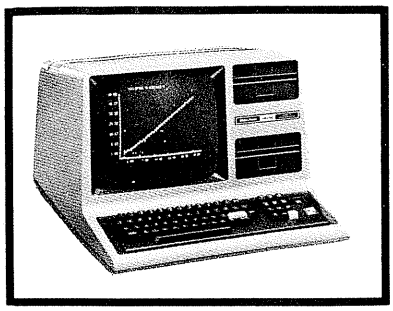
Height:	6 inches
Width:	15 inches
Depth:	10.5 inches
Weight:	17 pounds
Print speed:	2.8 inches per second
Step size:	Horizontal.005 inch Vertical.005 inch
Resolution:	200 increments per inch
Surface Area:	8½ x 11 inches
Plot Size:	7 x 9¼ inches

Signal Input Requirements:

EIA RS-232-C Serial Asynchronous

Preprogrammed Capabilities

- Relative and Absolute Coordinate System
- Straight Line Generator with solid or dash line
- Circle Generator
- Arc Generator
- Character Generator - 93 Characters
  - 4 Rotations
  - 9 Sizes
- Marker Generator - 6 sizes
- Pen Changes

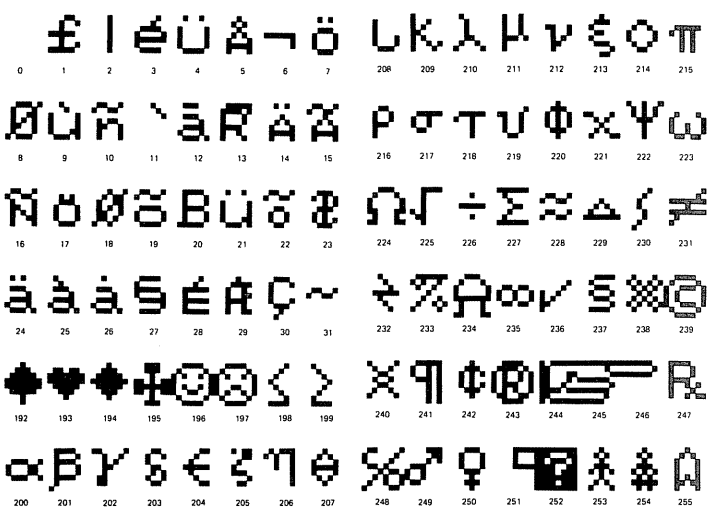


Model I/III

Product Line Manager's News

The Model III is more than a practical workhorse. To insure that you never forget to just have fun with your Model III we have tucked away in the Model III several likable little munchkins. These folks, along with the rest of Model III's special graphics character set, are easy to ferret out and fun to put to use.

Let's explore Model III's 96 special characters. We will look at how to access them in a minute but first here are Model III's special characters and the codes corresponding to each character:



What do you see? Codes 253 and 254 produce the little folks. Try 244 thru 246 to produce a pointing finger. Or 192, 193, 194 or 195 to help develop a card game. If someone answers your quiz incorrectly then use 197. Reward a correct answer with 196. There are all kinds of possibilities with the special characters, including even practical uses. You will see mathematical symbols like π (215), ∞ (235), ≠ (231), √ (225) and others. Or ¢ (242), c/o (248), and others. You will find a ✓ (236), a fat × (238), even the © (239), and ® (243) symbols.

So how do we use the codes to access the characters? To access them we will "switch" to the special character mode and/or poke into video RAM the appropriate codes corresponding to the desired character.

Here is a program to demonstrate how to poke the first 32 special characters into Video RAM and then poke the next 64:

```
1Ø CLS
15 PRINT CHR$(23)
2Ø FOR I=Ø TO 31
3Ø POKE 1536Ø+I*16,I
4Ø NEXT I
5Ø PRINT@64Ø, "READY FOR MORE?";
6Ø A$=INKEY$
7Ø IF A$="" THEN 6Ø ELSE 8Ø
8Ø PRINT CHR$(21)
9Ø FOR I=192 TO 255
1ØØ POKE 1536Ø+I*4, I
11Ø NEXT I
12Ø PRINT@64Ø,"PRESS ANY KEY TO CHANGE SIZE!";
13Ø IF INKEY$="" THEN 13Ø
14Ø IF N=8 THEN N=12 ELSE N=8
15Ø OUT 239, N
16Ø GOTO 13Ø
```

PRINT CHR$(21) in line 100 is the "switch" that activates the last 64 characters of the set. You could have also displayed the last 64 characters with a PRINT CHR$() statement. Don't forget, as you use the character set, that Model III allows poking of any of these characters into location 16419 to alter the cursor. Try a smiling face, inverse question mark or others.

MULTIPLICATION AID

Here is part of a letter from Dr. Robert Sherman of Melbourne, Florida. We have included the letter here, because his program "Wiz Math" uses the graphics "little people" quite effectively:

"I am happy to share with you one of my favorite programs. I developed "Wiz Math" for my daughter who had been having difficulty with her multiplication tables. The program is designed to ask the product of X and Y, where X and Y equal random numbers from two to ten. After ten such problems, with the appropriate "stroking" for correct answers, the user is asked whether he or she would like to do another ten problems. The coup de grace, however, is that whenever the user is having difficulty answering the question, he or she may press the digit "1" and a series of families appear. In other words, if the problem is 6 times 4, six stick figures in each of four families appear on the screen. The user then only has to count the number of little people and he or she has the correct answer. It was this last feature that turned the program into a game rather than a chore. I should add that my daughter is now quite good at her multiplication problems!"

Here is Dr. Sherman's program:

```
4 PRINT CHR$(21)
5 CLS
    : PRINT@461, "THIS GAME IS CALLED WIZMATH"
6 PRINT
7 PRINT "     LET'S SEE HOW MANY OF THESE MULTIPLICATION
    PROBLEMS"
8 PRINT "                    YOU CAN ANSWER."
9 FOR Z=1 TO 1ØØØ
    : NEXT Z
1Ø PRINT
12 PRINT "IF YOU'RE HAVING TROUBLE AND WANT TO SEE SOME
    FAMILIES"
13 PRINT "AT ANY TIME, JUST ENTER THE NUMBER 1."
    : FOR Z=1 TO 3ØØØ
    : NEXT Z
14 PRINT
    : CLS
15 FOR N=1 TO 1Ø
16 K=Ø
2Ø C=RND(9)
    : D=RND(9)
25 A=C+1
    : B=D+1
29 IF K=>2 GOSUB 2ØØØ ELSE P=Ø
3Ø PRINT "HOW MUCH IS "; A; " TIMES "; B
35 INPUT P
36 IF P=A*B GOTO 8Ø
38 IF P=1 GOTO 55
4Ø IF P<A*B AND P<>1 THEN PRINT "TOO LOW! TRY AGAIN!"
    : K=K+1
    : GOTO 29
5Ø IF P>A*B THEN PRINT "TO HIGH! TRY AGAIN!"
    : K=K+1
    : GOTO 29
55 FOR L=1 TO B
58 PRINT
6Ø FOR M=1 TO A
63 PRINT CHR$(253);
```

(Continued on page 12)

Model I/III (From Page 11)

```
65 NEXT M, L
7Ø K=Ø
72 PRINT
75 GOTO 29
8Ø IF N=1 PRINT "ABSOLUTELY CORRECT! YOU'RE FANTASTIC!"
9Ø IF N=2 PRINT "RIGHT ON THE BUTTON! YOU'RE REALLY OK!"
1ØØ IF N=3 PRINT "CORRECT! WAY TO GO, PAL!"
11Ø IF N=4 PRINT "YOU'RE RIGHT AGAIN! ANOTHER EINSTEIN!"
12Ø IF N=5 PRINT "JUST FABULOUS! YOU'RE RIGHT AGAIN!"
13Ø IF N=6 PRINT "I DON'T BELIEVE IT! RIGHT AGAIN!"
14Ø IF N=7 PRINT "RIGHT ON THE NOSE, BABY! YOU SURE KNOW
    YOUR STUFF!"
15Ø IF N=8 PRINT "RIGHT! YOU ARE OUT OF THIS WORLD!"
16Ø IF N=9 PRINT "WOW! YOU'RE RIGHT AGAIN! MARVELOUS!"
17Ø IF N=1Ø PRINT "CORRECT! YOU'RE A MATH WIZ! YOU GOT 1Ø
    PROBLEMS RIGHT!"
18Ø NEXT N
19Ø INPUT "DO YOU WANT TO PLAY AGAIN"; Y$
2ØØ IF Y$="Y" OR Y$="YES" THEN 5
21Ø PRINT CHR$(21)
    : CLS
    : END
2ØØØ PRINT "IF YOU'RE HAVING TROUBLE AND WANT TO SEE SOME
    FAMILIES"
2Ø1Ø PRINT "YOU CAN ALWAYS ENTER THE NUMBER 1."
2Ø2Ø RETURN
```

In the June, 1981 Newsletter I gave you a listing of part of a graphics program from our "Getting Started With TRS-80 BASIC" manual. Here is a letter about that program from William Keister, of Short Hills, New Jersey:

"You don't have to use the arrow keys, just because they are there, when you program sketching and drawing programs—particularly since there are no diagonal arrow keys. I use the keys clustered around the "5" key on the number pad of my Model I (eg. "8" for Up, "2" for Down, "9" for NE, etc.).

"For example, in the graphics drawing board program on page 3 in the June 1981 Newsletter (Ed.—You can also find this program on page 329 of 26-2107 "Getting Started With TRS-80 BASIC"), change the key definitions in lines 200-270 to the appropriate numbers:

```
WAS:                        NOW:

2ØØ UP$=CHR$(91)            2ØØ UP$="8"
21Ø DN$=CHR$(1Ø)            21Ø DN$="2"
22Ø LF$=CHR$(8)             22Ø LF$="4"
23Ø RT$=CHR$(9)             23Ø RT$="6"
24Ø NE$="Q"                 24Ø NE$="9"
25Ø NW$="W"                 25Ø NW$="7"
26Ø SE$="A"                 26Ø SE$="3"
27Ø SW$="S"                 27Ø SW$="1"
```

"While we are at it, we may as well change the write, autodraw and cursor mode controls to ".", "5", and "0" so that all the sketching controls are convenient at the number pad for one-finger drawing.

```
WAS:                            NOW:

28Ø WR$=CHR$(ASC("W")+32)       28Ø WR$="."
29Ø AD$=CHR$(ASC("A")+32)       29Ø AD$="5"
3ØØ CP$=CHR$(ASC("C")+32)       3ØØ CP$="Ø"
```

"This is an easy way to change the program with only a few simple edits. Try it—I think you will like it, particularly the "5" in the middle of the number pad that turns autodraw on and off.

"Since we are using numeric characters for drawing, we can use the VAL function and pick the X and Y increments out of arrays. Some people would say that this is a more elegant way to do the job. Here are some code modifications which will use the VAL function and convert the program to use the number pad:

DELETE lines 2ØØ-3ØØ and replace them with:

```
2ØØ 'DATA FOR X AN Y INCREMENTS
21Ø DATA -1, 1, Ø, 1, 1, 1, -1, Ø, Ø, Ø, 1, Ø, -1, -1, Ø,
    -1, 1, -1
22Ø 'LOAD INCREMENTS IN ARRAYS XD() AND YD()
23Ø FOR I=1 TO 9
    : READ XD(I), YD(I)
    : NEXT I
24Ø 'DEFINE WRITE AND CURSOR MODE CONTROL KEYS
25Ø WR$="."
    : CP$="Ø"
```

DELETE lines 1Ø6Ø-113Ø and replace them with:

```
1Ø6Ø 'AUTO ON/OFF (TAKE OUT "5" FIRST)
1Ø7Ø 'SKIP IF K$ NOT A NUMBER
1Ø8Ø IF K$="5" THEN A=-A
    : GOTO 124Ø
1Ø9Ø IF K$<"1" OR K$>"9" THEN 114Ø
11ØØ 'CHANGE CHAR. TO INTEGER
111Ø J=VAL(K$)
112Ø 'SET DRAW INCREMENTS DX, DY
113Ø DX=XD(J)
    : DY=YD(J)
    : GOTO 127Ø
```

DELETE line 115Ø (Line 1Ø6Ø does this now)

Try Mr. Keister's modifications, you may find you like the graphics program even better than before!

That is all until next month.

Model I/III Bugs, Errors and Fixes

Model III TRSDOS 1.3

There are now three versions of Model III TRSDOS 1.3. These versions are dated May 1, May 2, and July 1.

The following patches should be made to a May 1 disk inorder to upgrade it to a May 2 disk:

```
PATCH XFERSYS/CMD (ADD=548E,FIND=35ØØFD21,CHG=FD36ØØØ1)

PATCH *Ø (ADD=5Ø3B,FIND=467269,CHG=536174)
PATCH *Ø (ADD=5Ø44,FIND=31,CHG=32)

PATCH BASIC/CMD (ADD=58F8,FIND=F1,CHG=ØØ)
```

The first of these patches fixes a bug in XFERSYS which could prevent some of your files from being transferred properly.

The second set of patches changes the date of the release version.

The final patch corrects a bug in BASIC which caused the system to lockup if you repeatedly pressed the up-arrow key while you were at the top line in a BASIC program.

Any Model III TRSDOS 1.3 diskettes which are dated May 1 should have these four patches applied.

To upgrade a May 2 TRSDOS 1.3 diskette to a July 1 diskette, make the following patches (this includes any May 1 diskettes which you just upgraded to May 2):

```
PATCH *6 (ADD=585Ø,FIND=3A62,CHG=BF5F)
PATCH *6 (ADD=5FBE,FIND=2Ø697320616374,CHG=ØD116544C31C44)

PATCH *Ø (ADD=5Ø44,FIND=32,CHG=31)
PATCH *Ø (ADD=5Ø3A,FIND=2Ø536174204D,CHG=5765642Ø2Ø4A)
PATCH *Ø (ADD=5Ø4Ø,FIND=6179,CHG=756C)

PATCH *7 (ADD=579C,FIND=Ø955,CHG=3851)
PATCH *7 (ADD=5135,FIND=2Ø7468652Ø64,CHG=3F2ØØ33A7D4E)
PATCH *7 (ADD=513B,FIND=69736B657474,CHG=FE81CAØD55C9)
```

The first set of patches here corrects a problem with loading large files. The second set changes the date of the diskette, and the third set corrects an error when you formated a diskette in drive zero. (Continued on Page 13)

Model I/III Bugs, Etc. (From Page 12)

One additional patch which you should make to all three versions of TRSDOS 1.3 concerns the TRSDOS I/O (machine language) call to display the directory listing of all non-protected user files. This call (at HEX 4419) does not work correctly. The following two patches will cause this I/O call to function correctly:

```
PATCH *1Ø (ADD=4E2E, FIND=CD3E4B, CHG=CD8A5Ø)
PATCH *1Ø (ADD=5Ø8A, FIND=4469736B, CHG=4FC33E4B)
```

For Your Information:The TRSDOS 1.3 BASIC Interpreter "KILL" statement closes all open files before the KILL is executed.

With TRSDOS 1.1 and 1.2 it was possible that all the data on a diskette could be lost if an open file were killed. The procedure used in 1.3 prevents this from occuring. If you are going to KILL a file from a BASIC program, you will need to reOPEN any files you wish to continue using.

You should not KILL a file while you have a sequential file open. The sequential file will be closed by the KILL command, and you will lose your sequential file position and have to begin reading or writing from the beginning of the file. If you are using sequential files, KILLing files before opening (or after closing) needed sequential files will prevent problems with those files.

If you have a program which must OPEN and then KILL some files while leaving other files OPEN, the following patch may be made to prevent Disk BASIC from closing files before the KILL is issued:

```
PATCH BASIC/CMD (ADD=6ØDØ,FIND=CDE95C,CHG=ØØØØØØ)
```

WARNING: Once this patch has been made, you MUST take care that you do not KILL a file that is OPEN. To do so may destroy the diskette directory and render ALL programs and data on that disk useless!

Line Draw Routines

In the May, 1981 Microcomputer NEWS we published a small program titled "Line Draw" by Eddie Pont. Since then we also published a Drawing Board program in the June issue. These two programs have generated a significant amount of response, and we would like to share that response with you. We have made no attempt to take all of the material and integrate it into a single program. We thought you might like to "pick and choose" to create your own line draw routines.

In order to give you the proper perspective, here is Mr. Pont's original program:

```
    LINE DRAW

1 REM LINE DRAW
3 REM BY: EDDIE PONT*** JAN. '81
5 REM "U"=UP: "D"=DOWN
7 REM "R"=RIGHT: "L"=LEFT
1Ø CLS
    : ON ERROR GOTO 15Ø
2Ø B$="R"
3Ø SET(X,Y)
    : FOR T=1 TO 1ØØ
    : NEXT
4Ø A$=INKEY$
    : IF LEN(A$)>Ø GOTO 1ØØ
5Ø IF B$="U" THEN Y=Y-1
    : GOTO 3Ø
6Ø IF B$="D" THEN Y=Y+1
    : GOTO 3Ø
7Ø IF B$="R" THEN X=X+1
    : GOTO 3Ø
8Ø IF B$="L" THEN X=X-1
    : GOTO 3Ø
9Ø A$=""
    : GOTO 3Ø
1ØØ B$=A$
    : A$=""
11Ø GOTO 3Ø
15Ø IF ERR/2+1=5 GOTO 16Ø ELSE RESUME 4Ø
16Ø PRINT@ 896,"OUT OF RANGE";
17Ø A$=INKEY$
    : IF LEN(A$)=Ø GOTO 17Ø
18Ø PRINT@ 896,"           ";
19Ø B$=A$
    : A$=""
    : RESUME 5Ø
```

PROGRAM #1

Jim Gurd Dearborn Hts. Michigan

In your May '81 issue of Microcomputer News, you printed a letter by Eddie Pont called "Line Draw." I have modified the program slightly. Here is my program:

```
1 '*** LINE DRAW ***
3 '*** BY: EDDIE PONT ***
5 '*** MODIFIED BY: JIM GURD ***
1Ø CLS
    : ON ERROR GOTO 15Ø
2Ø B$= CHR$(9)
3Ø SET (X,Y)
4Ø A$=INKEY$
    : IF LEN(A$)>Ø THEN 1ØØ
5Ø IF B$=CHR$(91) THEN Y=Y-1
6Ø IF B$=CHR$(1Ø) THEN Y=Y+1
7Ø IF B$=CHR$(9) THEN X=X+1
8Ø IF B$=CHR$(8) THEN X=X-1
9Ø A$=""
    : GOTO 3Ø
1ØØ B$=A$
    : A$=""
11Ø GOTO 3Ø
15Ø IF ERR/2+1 <> 5 THEN ON ERROR GOTO Ø
    : RESUME
16Ø IF X<Ø THEN X=Ø
17Ø IF X>127 THEN X=127
18Ø IF Y<Ø THEN Y=Ø
19Ø IF Y>47 THEN Y=47
2ØØ RESUME
```

This program uses the right, left, up and down arrows to change the direction of the lines. The program will not let you go off the screen. This line can be added to the program if you need to slow down the display:

```
35 FOR T=Ø TO 1ØØ: NEXT T
```

Editor's note: Jim is twelve years old.

PROGRAM #2

Ara M. Baltayan New Haven, Connecticut

Here is a modification to Mr. Pont's program.

This simple modification adds more fun to the already interesting program of the original author by providing a means of jumping up, down, right and left and continuing in any direction from that point on. It also permits putting isolated dots across the screen.

Incidentally, I had to add a " on line 180 which was a typographical omission. No one is perfect after all.

```
1 REM LINE DRAW
3 REM BY: EDDIE
4 REM MODIFIED BY: ARA M. BALTAYAN, P.E. *** MAY '81
7 REM "R"=RIGHT: "L"=LEFT
8 REM "A"=JUMP RIGHT: "B"=JUMP LEFT
1Ø CLS
    : ON ERROR GOTO 15Ø
2Ø B$= "R"
3Ø SET (X,Y)
    : FOR T=1 TO 1ØØ
    : NEXT
4Ø A$=INKEY$
    : IF LEN(A$)>Ø GOTO 1ØØ
5Ø IF B$="U" THEN Y=Y-1
    : GOTO 3Ø
6Ø IF B$="D" THEN Y=Y+1
    : GOTO 3Ø
7Ø IF B$="R" THEN X=X+1
    : GOTO 3Ø
```

(Continued on Page 14)

Line Draw (From Page 13)

```
8Ø IF B$="L" THEN X=X-1
     : GOTO 3Ø
82 IF B$="A" THEN X=X+1,
     : GOTO 3Ø
84 IF B$="B" THEN X=X-1,
     : GOTO 3Ø
86 IF B$="C" THEN Y=Y+1,
     : GOTO 3Ø
88 IF B$="E" THEN Y=Y-1,
     : GOTO 3Ø
9Ø A$=""
    : GOTO 3Ø
1ØØ B$=A$
    : A$=""
11Ø GOTO 3Ø
15Ø IF ERR/2+1 = 5 GOTO 16Ø ELSE RESUME 4Ø
16Ø PRINT@ 896, "OUT OF RANGE";
17Ø A$=INKEY$
     : IF LEN(A$)=Ø GOTO 17Ø
18Ø PRINT@ 896, "            ";
19Ø B$=A$
     : A$=""
     : RESUME 5Ø
```

PROGRAM #3

G. M. Fitzwater Santa Cruz, California

In your May issue you took the time to print an enormous amount of information concerning printer-driver routines and formats for the various models which I am still digesting. Thank you for this valuable reference information.

The last page of that newsletter contained a user note from Eddie Pont called "Line Draw" which was, frankly, surprising in that the techniques used were so artificial and awkward. Much shorter and more useful routines have been available in published materials for at least 18 months. I have myself pulled together a line drawing routine, using the keyboard vector table, that not only accomplishes everything in "Line Draw," but will erase when the shift key is depressed. Diagonals may be drawn by depressing two keys, and arrows instead of letters are used. Key depression recognition is virtually instantaneous without the sometimes annoying response profile of INKEY$.

```
2ØØØØ CLS
    : X=63
    : Y=23
2ØØ1Ø W=PEEK(144ØØ)
    : FOR I=1 TO 2Ø
    : NEXT
2ØØ2Ø IF W AND 8 THEN IF Y>Ø THEN Y=Y-1
2ØØ3Ø IF W AND 16 THEN IF Y<47 THEN Y=Y+1
2ØØ4Ø IF W AND 32 THEN IF X>Ø THEN X=X-1
2ØØ5Ø IF W AND 64 THEN IF X<126 THEN X=X+1
2ØØ6Ø SET(X,Y)
    : IF PEEK(14464) THEN RESET(X,Y)
2ØØ7Ø GOTO 2ØØ1Ø
```

Editor: For those of you who may be unfamiliar with the technique that Mr. Fitzwater is using, address locations 14400 and 14464 are located in keyboard memory. By looking at these locations, and performing the compares in program lines 20020 to 20050, you can determine if one or more of the arrow keys is being pressed. One of the advantages of this technique is that you can press more than one arrow key and easily draw diagonal lines.

PROGRAM #4

Thomas Brown Alexander City, Alabama

I recently read an article in the newsletter where the user draws on the TRS-80 screen. I have found that this program is very hard to control.

I have developed a program that allows you to draw on the screen with much more control. My program can also move along diagonals where the other program only moves in horizontal or vertical directions.

In this program, you use the arrow keys to move a blinking dot around the screen leaving a trail. If the 'C' key is pressed, the drawing mode is changed. For example, if the dot is setting points and the 'C' key is pressed, then the dot will reset any points it crosses. If the 'C' key is pressed when the dot is resetting points, then the mode will change back to setting the points.

```
1Ø CLS
    : A=64
    : B=23
2Ø IF PEEK(15168) = 32 THEN A=A-1 ELSE IF PEEK(15168) = 64 THEN
    A=A+1
3Ø IF PEEK(15168) = 8 THEN B=B-1 ELSE IF PEEK(15168) = 16 THEN
    B=B+1
4Ø IF PEEK(15168) = 72 THEN A=A+1
    : B=B-1 ELSE IF PEEK(15168) = 8Ø THEN A=A+1
    : B=B+1
5Ø IF PEEK(15168) = 4Ø THEN A=A-1
    : B=B-1 ELSE IF PEEK(15168) = 48 THEN A=A-1
    : B=B+1
6Ø IF A>125 THEN A=A-1 ELSE IF A<2 THEN A=A+1
7Ø IF B<2 THEN B=B+1 ELSE IF B>45 THEN B=B-1
8Ø RESET(A,B)
9Ø FOR T=1 TO 5Ø
    : NEXT
1ØØ SET(A,B)
11Ø A$=INKEY$
12Ø IF A$="C" THEN GOSUB 15Ø
13Ø IF O=1 THEN RESET(A,B)
14Ø GOTO 2Ø
15Ø IF O=1 THEN O=Ø
    : RETURN
16Ø IF O=Ø THEN O=1
    : RETURN
17Ø ' THOMAS BROWN
18Ø ' 15Ø4 EAST 4TH STREET
19Ø ' ALEX CITY, ALABAMA 35Ø1Ø
```

PROGRAM #5

Stuart Rowland Muldrow, Oklahoma

In the past the newsletter has published programs to draw on the computer. But, to date, these programs have been long and slow. Here are a couple of programs I've written that should spice up computer graphics on the Models I and III.

This first program uses SET/RESET for graphics:

```
1Ø CLS
    : INPUT"STARTING POINT (X,Y)"; X, Y
    : IF X<Ø OR Y<Ø OR X>127 OR Y>47 THEN 1Ø ELSE CLS
2Ø SET(X,Y)
    : X1=X
    : Y1=Y
3Ø A=PEEK(144ØØ)
    : IF A=Ø THEN RESET(X,Y)
    : GOTO 2Ø
    : ELSE IF A=13Ø THEN CLS
    : GOTO 2Ø
    : ELSE IF A>1ØØ THEN Z=1 ELSE Z=Ø
4Ø IF A AND 8 THEN Y=Y-1 ELSE IF A AND 16 THEN Y=Y+1
5Ø IF A AND 32 THEN X=X-1 ELSE IF A AND 64 THEN X=X+1
6Ø IF X<Ø OR X>127 THEN X=X1
7Ø IF Y<Ø OR Y>47 THEN Y=Y1
8Ø IF Z=Ø THEN 2Ø ELSE RESET(X,Y)
    : X1=X
    : Y1=Y
    : GOTO 3Ø
```

This second program lets you use 10 different characters to draw with. You can change characters while you are drawing, Enlarge, Reduce, and Invert video. Both programs use the arrow keys to draw.

```
1Ø CLS
    : CLEAR 3ØØ
    : DIM A(9)
    : FOR I=1 TO 9
```

(Continued on Page 15)

Line Draw (From Page 14)

```
2Ø PRINT "CHARACTER CODE FOR ";I;
    : INPUT A(I)
    : IF A(I)<1 OR A(I)>255 THEN 2Ø ELSE NEXT
3Ø CLS
    : INPUT"STARTING POINT (Ø-63, Ø-15)"; X,Y
    : IF X<Ø OR Y<Ø OR X>63 OR Y>15 THEN 3Ø ELSE P=(Y*64)+X
    : P=P+15359
    : CLS
    : PF=A(Ø)
    : POKE P, PF
4Ø GOSUB 6Ø
    : I$=INKEY$
    : IF I$="" THEN 4Ø ELSE IF I$="E" AND T=Ø THEN PRINT
    CHR$(23);
    : T=1
    : GOTO 4Ø
    : ELSE IF I$="R" AND T=1 THEN PRINT CHR$(28);
    : T=Ø
    : GOTO 4Ø
    : ELSE IF I$="I" THEN GOSUB 11Ø
    : GOTO 4Ø
5Ø IF ASC(I$)>47 AND ASC(I$)<58 THEN Z=VAL(I$)
    : PF=A(Z)
    : GOTO 4Ø
    : ELSE 4Ø
6Ø F=PEEK(14ØØ)
    : IF F=Ø THEN RETURN ELSE IF F=128 THEN CLS ELSE IF F=2 THEN
    3Ø ELSE IF F=13Ø THEN RUN
7Ø P1=P
    : IF F AND 8 THEN P=P-64 ELSE IF F AND 16 THEN P=P+64
8Ø IF F AND 32 THEN P=P-1 ELSE IF F AND 64 THEN P=P+1
9Ø IF P<15360 OR P>16383 THEN P=P1
1ØØ POKE P, PF
    : RETURN
11Ø FOR I=15360 TO 16383
    : Q=PEEK(I)
    : IF Q=32 THEN POKE I, 191
    : NEXT
    : ELSE IF Q<128 OR Q>191 THEN NEXT
    : ELSE Q=(159.5-Q)+159.5
    : POKE I,Q
    : NEXT
12Ø RETURN
```

PROGRAM #6

Ross Leventhal Sun Valley, Idaho

In the May, 1981 issue of the Newsletter, Eddie Pont had a draw program that I modified into a game. My friends compete with each other and try to get high scores. The most points possible is 5625. The object is to fill up the most space possible without hitting a filled space. I hope you will enjoy the game.

```
1 CLS
    : RANDOM
2 PRINT@4Ø1, "TO CONTROL THE LINE USE THE KEYPAD:"
3 PRINT@465, "4=LEFT    6=RIGHT"
4 PRINT@529, "8=UP      2=DOWN"
5 PRINT@596, "5=REAPPEARING AT A RANDOM SQUARE"
8 GOSUB 5ØØ
9 FOR T=Ø TO 13ØØ
    : NEXT
    : PRINT@4Ø1, "                                        ";
    : REM THE NEXT 4 PRINT@'S ARE TO CLEAR THE SCREEN SO MAKE
    SURE THERE ARE THE SAME NUMBER OF SPACES AS THERE WERE
    CHARACTERS IN THE WORDING OF LINES 2,3,4, AND 5
1Ø PRINT@465, "                ";
11 PRINT@529, "               ";
12 PRINT@596, "                                    ";
14 Z=Ø
15 X=62
    : Y=22
2Ø B$="6"
3Ø Z=ABS(Z)+1
    : SET(X,Y)
    : IF Z<1ØØ GOTO 36Ø
    : IF Z(>1ØØ) AND (<3ØØ) GOTO 37Ø
    : IF Z(>3Ø1) AND (<6ØØ) GOTO 38Ø
    : IF Z>6Ø1 GOTO 4Ø
4Ø A$=INKEY$
    : IF LEN(A$)>Ø GOTO 1ØØ
45 IF B$="5" THEN GOTO 21Ø
5Ø IF B$="8" THEN Y=Y-1
    : IF POINT(X,Y)=-1 GOTO 3ØØ ELSE GOTO 3Ø
6Ø IF B$="2" THEN Y=Y+1
    : IF POINT(X,Y)=-1 GOTO 3ØØ ELSE GOTO 3Ø
7Ø IF B$="6" THEN X=X+1
    : IF POINT(X,Y)=-1 GOTO 3ØØ ELSE GOTO 3Ø
8Ø IF B$="4" THEN X=X-1
    : IF POINT(X,Y)=-1 GOTO 3ØØ ELSE GOTO 3Ø
9Ø A$=""
    : GOTO 3Ø
1ØØ B$=A$
    : A$=""
11Ø GOTO 3Ø
21Ø X=RND(126)
    : Y=RND(46)
215 IF POINT(X,Y)=-1 GOTO 3ØØ
22Ø SET(X,Y)
    : GOSUB 26Ø
    : RESET(X,Y)
    : GOSUB 26Ø
    : SET(X,Y)
23Ø B$="6"
    : GOTO 3Ø
26Ø FOR K=Ø TO 5Ø
    : NEXT
27Ø RETURN
3ØØ V=ABS(V)+1
    : CLS
    : PRINT@477, "SCORE";
    : PRINT@541, Z;
    : IF V>9 GOTO 31Ø ELSE GOTO 4ØØ
31Ø PRINT@578, "WOULD YOU LIKE TO TRY AGAIN?";
    : INPUT A$
32Ø IF A$="YES" THEN GOTO 7ØØ
33Ø PRINT "HOPE YOU LIKED THE GAME.  SEE YOU NEXT TIME."
34Ø GOTO 33Ø
35Ø FOR T=Ø TO 5Ø
    : NEXT
    : GOTO 4Ø
36Ø FOR T=Ø TO 25
    : NEXT
    : GOTO 4Ø
37Ø FOR T=Ø TO 1Ø
    : NEXT
    : GOTO 4Ø
38Ø FOR T=Ø TO 5
    : NEXT
    : GOTO 4Ø
4ØØ GOSUB 5ØØ
    : PRINT@477, "       ";
    : PRINT@541, "       ";
    : GOTO 14
5ØØ FOR A= Ø TO 127
    : FOR B=Ø TO 47 STEP 47
    : SET(A,B)
    : NEXT B,A
51Ø FOR A=Ø TO 127 STEP 127
    : FOR B=Ø TO 47
    : SET(A,B)
    : NEXT B,A
52Ø RETURN
7ØØ V=Ø
    : PRINT@578, "                              ";
    : GOSUB 5ØØ
    : GOTO 4ØØ
```

PROGRAM #7

Mark Jeghers Stockton, California

In your May, 1981 Newsletter, you presented a line drawing program by Eddie Pont. Well, one good routine deserves another so here is 'Draw II'. This program will turn any Model I Level II or Model III (without disk) into a glorified Etcha-sketch. Please note that this program was not designed to work with disk systems. Special routines use the USR(0) function with two machine language routines neatly wrapped in a string variable (U$). VARPTR(U$) provides entry points for the routines (white-out and flip picture).

(Continued on Page 16)

Line Draw (From Page 15)

Features of this program are as follows:

Speed control-keys '1' (fast) through '9' (slow)—helpful for careful drawings
Arrow keys control the direction of lines—'@' key to stop line—easier to use than U-D-R-L keys . . .
3 line modes—'D' key: Draw line
'E' key: Erase line
'X' key: Draw where erased and erase where drawn.
3 display controls—'CLEAR' key: clears the screen
'W' key: WHITES OUT SCREEN!
'F' key: FLIPS SCREEN! (inverts shades)
2 dot controls—'Z' key: set line cursor to graphics location (0,0)—
'M' key: move cursor around screen without messing up your picture
'R' key: WRAP-AROUND, toggles on or off.

If you play with this for a while you will see the potential for unlocking the artistic dreams of all the modern-day Van Gogh's out there . . .

```
Ø '     D R A W
1 ' FOR 16K LEVEL II OR 16K MODEL III BASIC
1Ø CLEAR 75
    : DEFINT A-Z
    : CLS
    : M=1
    : C=Ø
    : X=63
    : Y=23
    : P=16526
    : R=4
    : Q$=" "
    : U$=""
    : FOR I=1 TO 37
    : READ A
    : U$=U$+CHR$(A)
    : NEXT
2Ø IF VAL(Q$)<>Ø THEN R=VAL(Q$)-1
    : GOTO 5ØØ
3Ø IF Q$="R" AND WR=1 THEN WR=Ø
    : ELSE IF Q$="R" AND WR=Ø THEN WR=1
    : ELSE IF Q$="@" THEN C=Ø
    : ELSE IF Q$="[" THEN C=1
    : ELSE IF Q$=CHR$(9) THEN C=2
    : ELSE IF Q$=CHR$(1Ø) THEN C=3
32 IF Q$=CHR$(8) THEN C=4
4Ø V=VARPTR(U$)
    : IF Q$="W" THEN POKE P, PEEK(V+1)
    : POKE P+1, PEEK(V+2)
    : A=USR(Ø)
    : ELSE IF Q$="F" THEN POKE P, PEEK(V+1)+14
    : POKE P+1, PEEK(V+2)
    : A=USR(Ø)
42 IF Q$="Z" THEN C=Ø
    : M=Ø
    : X=Ø
    : Y=Ø
    : ELSE IF ASC(Q$)=31 THEN CLS
    : ELSE IF Q$="M" THEN M=Ø
    : ELSE IF Q$="D" THEN M=1
    : ELSE IF Q$="E" THEN M=2
    : ELSE IF Q$="X" THEN M=3
5Ø IF C=Ø THEN 8Ø ELSE ON C GOTO 51, 52, 53, 54
51 Y=Y-1
    : GOTO 6Ø
52 X=X+1
    : GOTO 6Ø
53 Y=Y+1
    : GOTO 6Ø
54 X=X-1
    : GOTO 6Ø
6Ø ON WR+1 GOTO 61, 63
61 IF X>127 THEN X=127
    : ELSE IF X<Ø THEN X=Ø
    : ELSE IF Y>47 THEN Y=47
    : ELSE IF Y<Ø THEN Y=Ø
62 GOTO 64
63 IF X>127 THEN X=Ø
    : ELSE IF X<Ø THEN X=127
    : ELSE IF Y>47 THEN Y=Ø
    : ELSE IF Y<Ø THEN Y=47
64 IF M=Ø THEN 8Ø ELSE ON M GOTO 65, 66, 67
65 SET(X,Y)
    : GOTO 8Ø
66 RESET(X,Y)
    : GOTO 8Ø
67 IF (POINT(X,Y)) THEN 66 ELSE 65
8Ø IF (POINT(X,Y)) THEN RESET(X,Y)
    : GOSUB 999
    : SET(X,Y)
    : ELSE SET(X,Y)
    : GOSUB 999
    : RESET(X,Y)
1ØØ FOR I=Ø TO 1Ø*R*R
    : NEXT
5ØØ Q$=INKEY$
    : IF Q$="" THEN Q$=" "
51Ø GOTO 2Ø
6ØØ DATA 33, Ø, 6Ø, 54, 191, 17, 1, 6Ø, 1, 255, 3, 237, 176,
    2Ø1, 33, Ø, 6Ø, 1, Ø, 4, 126, 254, 32, 32, 2, 62, 128,
    238, 63, 119, 35, 11, 12Ø, 177, 32, 24Ø, 2Ø1
999 FOR I=1 TO 5
    : NEXT
    : RETURN
```

Program # 8

Ofer Zikel Quincy, Illinois

I own a TRS-80 Model III, and I have been receiving the Newsletter for the past year or so. All of the line drawing programs use INKEY$, which is hard to control. Well, I have a line draw program that uses PEEK, which is a lot easier to control because when you lift your finger, the line stops automatically. Here is my program:

```
1Ø ' LINE DRAW USING PEEK BY OFER G. ZIKEL
2Ø ' THIS PROGRAM WILL WORK ON THE MODELS I AND III
3Ø ' THE PROGRAM FINDS OUT WHAT KEY IS BEING DEPRESSED
4Ø ' BY PEEKING ADDRESS 144ØØ
5Ø ' FORWARD ARROW = 64
6Ø ' BACKWARD ARROW = 32
7Ø ' UP ARROW = 8
8Ø ' DOWN ARROW = 16
9Ø ' WHEN A COMBINATION OF KEYS ARE BEING PRESSED THE
1ØØ ' COMPUTER WILL DRAW DIAGONALLY
1Ø1 ' WHEN YOU WANT TO ERASE, OR MOVE THE CURSOR WITHOUT
1Ø2 ' DRAWING, PRESS THE (SPACE-BAR), WHEN YOU WANT TO
1Ø3 ' CLEAR THE SCREEN PRESS (CLEAR).
11Ø '********** LINE DRAW **********
12Ø CLS
13Ø Q=PEEK(144ØØ)
135 ' CHECK FOR (SPACE-BAR) PRESSED
14Ø IF Q>=128 THEN Z=128 ELSE Z=Ø
15Ø IF Q=64+Z THEN X=X+1
16Ø IF Q=32+Z THEN X=X-1
17Ø IF Q=16+Z THEN Y=Y+1
18Ø IF Q=8+Z THEN Y=Y-1
19Ø IF Q=64+16+Z THEN X=X+1
    : Y=Y+1
2ØØ IF Q=64+8+Z THEN X=X+1
    : Y=Y-1
21Ø IF Q=32+16+Z THEN X=X-1
    : Y=Y+1
22Ø IF Q=32+8+Z THEN X=X-1
    : Y=Y-1
225 IF Q=2+Z THEN CLS
23Ø IF X>127 THEN X=127
24Ø IF Y>47 THEN Y=47
25Ø IF X<Ø THEN X=Ø
26Ø IF Y<Ø THEN Y=Ø
27Ø SET(X,Y)
28Ø IF Z=128 THEN RESET(X,Y)
29Ø GOTO 13Ø
```

Education

Educational Products News

INTRODUCING CHAMPS ™ : A Football Scouting Program By Luanne Kelly

"Computerized football" usually refers to a video game for family fun, but Radio Shack® will soon offer a unique computerized football scouting program that brings the TRS-80® to real high school and college football teams. **CHAMPS ™ : A Football Scouting Program** was designed by two high school football coaches to store and to analyze large amounts of information about the offensive tendencies of other teams, and to print a variety of detailed tendency reports when you ask for them.

Coaches J. Richard Dewey and Mathew L. Rauh, Jr., designed the system so that both the computer applications and the non-computerized features of CHAMPS would be of great professional interest to high school and to college football coaches alike. The computer program is extremely versatile, with many types of scouting reports to select from, and the computer eliminates almost all of the long hours that coaches would otherwise have to spend in manually sorting scouting information. The non-computerized features of the overall CHAMPS scouting system are designed to help teams increase their efficiency in the actual process of collecting scouting information.

Coaches Dewey and Rauh developed CHAMPS while they were coaching at Apple Valley High School in Apple Valley, Minnesota (near Minneapolis). First, they organized the collection of tendency data by designing a special Scouting Manual that their scouts could take to each game. The CHAMPS Scouting Manual is included with the CHAMPS computer program package, and is filled with pages whose special formats help the scouts know exactly what information they should record.

Reference pages are also included to help the scouts identify formations, and to assist them in describing 12 different features of each play run by the scouted team's offense (these described features are what the computer uses to print out the many varied offensive tendency reports that are available through CHAMPS).

Although the kicking game report and the defensive report are not computerized, the Scouting Manual guides the scouts in producing a well-organized and meaningful written record of the scouted team's defense and kicking game. The Scouting Manual also includes Preview pages for the coaches to fill out before each scouting assignment, to help the scouts get to know the team they will be scouting.

With the Scouting Manual, Coach Rauh and Coach Dewey found that they were able to use even inexperienced scouts, increase the amount of scouting information that they received, and make sure that the information was recorded in a well-organized format. To handle the large amount of data, and to produce the comprehensive, detailed reports that were now available from their wealth of information, Dewey and Rauh next turned to the computer.

To work with CHAMPS, you need a 48K Model III or Model I Disk system, any 132-column Radio Shack line printer, and the software package, which will include the CHAMPS computer program, ten copies of the CHAMPS Scouting Manual, and an instruction manual explaining the use of the program.

Once the CHAMPS program is loaded into the TRS-80 and is running, the computer will ask you to type in the name of the team that you are scouting. Once you have typed and entered the name, you will see a list of eight options for working with that team's scouting file.

```
                    CHAMPS OPTIONS

1 - ADD ..... ENTER SCOUTING DATA INTO A FILE
2 - LIST ..... LIST ALL DATA IN THE FILE BY PLAY
3 - CHANGE ..... CHAFE DATA IN A FILE
4 - AUTOMATIC .. PRINT OUT PRE-DESIGNED TENDENCY
                 CHARTS AND LISTINGS
5 - MORE INFO .. PRINT OUT ADDITIONAL COMPOSITE
                 TENDENCY LISTINGS
6 - PERCENT .... CALCULATE PERCENT OCCURRENCE OF
                 ITEMS IN A SPECIFIED DATA CATEGORY
7 - SORT ..... SORT A SPECIFIED DATA CATEGORY
8 - END ..... END THE PROGRAM

ENTER SELECTION: __

CHAMPS Main Menu
```

You can ADD information to the file, or CHANGE pieces of information that are already being stored. Five of the options allow you to use the information that you've stored to print tendency reports on the line printer. These report-printing options include the LIST option, which lists all of the stored information about a team, and the AUTOMATIC option, which allows you to print reports in as many as ten different pre-designed formats.

The MORE INFO., PERCENT, and SORT options allow you to print listings that answer the specific tendency questions that you choose to ask. In the MORE INFO. option, you define a situation on the field, and the computer will tell you what plays the scouted team has run in that situation, against which opponents the plays were run, and how many yards were gained or lost on each play.

```
ALL PLAYS 1ST AND 10/RESULTING IN 7 TO 10 YD GAIN


16 H TB + 9 R
02 OPT TB + 7 R
16 H TB +10 R
16 DIVE + 7 L
12 VEER + 7 L

MORE INFO option.  Example: 12 VEER + 7 L; 12 VEER is
the play name, + 7 is the yards gained or lost, L is the
identifying initial of the opponent
```

MORE INFO option. Example: 12 VEER + 7 L; 12 VEER is the play name, + 7 is the yards gained or lost, L is the identifying initial of the opponent

In the PERCENT option you also define a situation, and the computer will give you the percent occurrence (in that situation) of certain actions or situations that interest you. For example, you could use the PERCENT option to find out what percentage of the plays selected by the scouted team were runs, what percentage were passes, and what percentage were kicks, given a particular field position as the situation.

```
PERCENT OF RUN/PASS/KICK (-41 TO +35)


  ITEM                PERCENT


1     RUN               78.00
2     PASS              20.00
3     KICK               2.00
```

(Continued on Page 18)

Education (From Page 17)

The SORT option will sort all information of a specified type; for example, a "Down and Distance Sort" will tell you which plays were run on 1st and 10, which on 1st and more than 10, which on 1st and less than 10, which on 2nd and long, etc. The last of the eight options is the END option, which terminates the program.

Now let's briefly explore the ten different kinds of reports available through the AUTOMATIC option.

```
                    CHAMPS AUTOMATIC OPTION
DO YOU WANT (Y/N):
A LIST OF ALL HOT LINE INFORMATION BY PLAY TYPE?
A LIST OF ALL FORMATION COMBINATIONS USED BY YOUR OPPONENT?
A DOWN & DISTANCE ATTACK SHEET (BY FORMATION)?
A BACK ATTACK SHEET (BY FORMATION)?
A HASH ATTACK SHEET (BY FORMATION)?
A GOAL LINE ATTACK SHEET (BY FORMATION)?
THE VERTICAL FIELD POSITION TENDENCIES OF YOUR OPPONENT (A
LIST OF PLAYS BY VFP)?
A SERIES CHART?
A LIST OF ALL PLAYS THAT GAINED FOUR OR MORE YARDS?
A LIST OF ALL TOUCHDOWN PLAYS?
```

Automatic Option Screen

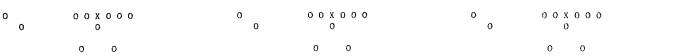
The first, "All Hot Line Information by Play Type," takes the play name, the yards gained or lost, and the initial of the scouted team's opponent for each play, and sorts this information by play type (run, pass, or kick). The second report, "All Formation Combinations Used by Your Opponent," is a directory of all of the different combinations of initial formation, adjustment to formation, and final formation that the scouted team has used in the games you've scouted.

```
          * ALL FORMATION COMBINATIONS USED BY YOUR OPPONENT *
OPPONENT:
YEAR:
  WIDE SLOT RIGHT - 3 / NO ADJUSTMENT TO FORMATION / WIDE SLOT RIGHT OPEN - 3
        WIDE SLOT RIGHT OPEN - 3 / TAILBACK FLY (WEAK) / PRO LEFT OPEN
  WIDE SLOT LEFT OPEN -3 / NO ADJUSTMENT TO FORMATION / WIDE SLOT LEFT OPEN -3
         PRO RIGHT OPEN / NO ADJUSTMENT TO FORMATION / PRO RIGHT OPEN
          PRO LEFT OPEN / NO ADJUSTMENT TO FORMATION / PRO LEFT OPEN
                      T / NO ADJUSTMENT TO FORMATION / T
            FIELD GOAL / NO ADJUSTMENT TO FORMATION / FIELD GOAL
           EXTRA POINT / NO ADJUSTMENT TO FORMATION / EXTRA POINT
                  PUNT / NO ADJUSTMENT TO FORMATION / PUNT
```

Automatic Option Report #2

The third, fourth, fifth, and sixth reports of the AUTOMATIC option are: the "Down and Distance Attack Sheet" (which develops a composite of down and distance tendencies), the "Back Attack Sheet" (which reveals any tendencies related to carriers or receivers), the "Hash Attack Sheet," (which reveals the short side/wide side tendencies of the scouted team), and the "Goal Line Attack Sheet" (which highlights the action from the +7 yard line to the goal line). The third through sixth reports are presented by formation; within each report, a separate tendency chart is printed for each formation combination used by the scouted team.

HASH ATTACK SHEET (BY FORMATION)



WIDE SLOT LEFT OPEN -3 / NO ADJUSTMENT TO FORMATION / WIDE SLOT LEFT OPEN -3

	LT FLANK	LE	LT	LG	C	RG	RT	RE	RT FLANK	PASSES
LEFT HASH		18 VEER +7 L						12 VEER -.5 R	02 OPT TB +18 L	
MIDDLE HASH	08 OPT TB +13 L			16 DIVE + 3 L		24 TRAP - 1 R	53 + 1 R	12 VEER + 2 R		
						24 TRAP FUM L		12 VEER + 7 L		
						24 TRAP + 1 L		12 VEER + 5 L		
RIGHT HASH	16 H TB +11 R	08 OPT QB + 5 L		16 DIVE + 3 R		24 TRAP + 4 R		12 VEER + 4 R	01 PITCH + 7 R	14 Z CIR INC R
	16 H TB + 9 R			16 DIVE + 3 L		24 TRAP +10 R		12 VEER + 6 L		79 DIG +17 R
	16 H TB +10 R			16 DIVE + 5 L		24 TRAP + 2 L		12 VEER + 7 L		79 Y DRAG +15 L
	16 H TB - 1 L			16 DIVE + 7 L		24 TRAP + 3 L		12 VEER - 1 L		79 Y DRAG +12 L
				16 DIVE + 2 L		24 TRAP - 2 L				79 CURL INC L
						24 TRAP + 2 L				79 FLOOD INT L
	9	8	7	6	5	4	3	2	1	10-19

TOTAL NUMBER OF RUNS 33
TOTAL NUMBER OF PASSES 6

OPPONENT:

YEAR:

COACHES' NOTES:

The seventh of the AUTOMATIC reports is the "Vertical Field Position Tendencies" report, a list of plays sorted by the vertical field position. (Continued on Page 19)

```
                * VERTICAL FIELD POSITION TENDENCIES *

OPPONENT:

YEAR:

(-0 TO -19)          (-20 TO -40)         (-41 TO +35)         (+34 TO +20)         (+19 TO +8)          (+7 TO GOAL LINE)
-----------          ------------         ------------         ------------         -----------          -----------------

14 DIVE + 3 R        14 DIVE + 3 R        14 DIG INC R         14 Z CIR +15 R       07 POWER + 5 R       07 POWER + 5 R
53 + 1 R             18 VEER +11 R        02 OPT QB + 8 R      16 READ + 2 R        03 POWER + 4 R       QB SNEAK + 1 R
24 TRAP - 1 R        16 READ + 3 R        16 DIVE + 3 R        71 CURL +12 R        QB SNEAK + 2 R       EX POINT GD R
PUNT +36 R           14 DIVE + 5 R        12 VEER + 2 R        02 OPT QB + 6 R      18 VEER + 6 R        18 VEER + 3 R
07 POWER + 3 L       18 VEER + 6 R        16 H TB +11 R        09 PITCH + 6 R       02 OPT TB + 7 R      18 VEER + 3 R
03 POWER + 5 L       02 OPT QB +20 R      08 OPT TB FUM R      75 DRAW + 5 R        18 VEER - 3 R        EX POINT GD R
                     14 DIVE + 3 R        14 Z CIR INC R       02 OPT PASS +30 R    71 CURL INC R        18 VEER + 6 R
                     12 VEER + 4 R        24 TRAP + 4 R        71 SPOT +19 R        71 TB SC LT + 6 R    EX POINT NG R
                     24 TRAP +10 R        01 PITCH + 7 R       79 DIG +17 R         FIELD GOAL NG R      09 PITCH + 2 R
                     14 DIVE + 3 L        16 H TB + 9 R        18 VEER +17 R        02 OPT TB + 6 R      54 SP POWER + 2 R
                     08 OPT TB +13 L      71 OUT INC R         QB SNEAK + 2 L       07 POWER + 4 R       07 POWER + 2 R
                     16 DIVE + 5 L        09 PITCH + 7 R       16 DIVE + 3 L        03 POWER + 4 L       QB SNEAK + 1 R
                     16 DIVE + 7 L        26 TRAP + 3 R        18 VEER + 5 L        07 POWER + 3 L       EX POINT GD R
                     24 TRAP FUM L        18 VEER QB + 2 R     26 TRAP + 4 L        07 POWER + 2 L       03 POWER + 4 R
                     14 DIVE + 6 L        79 FLOOD +12 R       02 OPT TB +14 L      QB SNEAK + 2 L       03 POWER + 2 R
                     18 VEER + 5 L        26 TRAP + 4 R        QB SNEAK + 1 L       54 SP POWER + 0 L    EX POINT GD R
                     26 TRAP + 1 L        18 VEER + 7 R        12 VEER - 1 L        09 PITCH + 3 L       02 OPT TB + 5 L
                     PUNT +33 L           02 OPT TB +20 R      79 Y DRAG +12 L      56 SP POWER + 2 L    EX POINT NG L
                     12 VEER + 6 L        16 H TB +10 R        18 VEER + 2 L        02 OPT TB +18 L      03 POWER + 1 L
                     24 TRAP + 1 L        18 VEER + 5 R                             ------------------   01 PITCH + 5 L
                     PUNT +41 L           12 VEER + 5 R                             (+7 TO GOAL LINE)    EX POINT NG L
                     18 VEER + 3 L        14 DIVE + 4 L                             -----------------    QB SNEAK + 1 L
                     12 VEER + 5 L        ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  07 POWER + 5 R       EX POINT GD L
                     26 TRAP + 3 L        14 H TB + 7 L                             QB SNEAK + 1 R
                     02 OPT TB +10 L      24 TRAP + 2 L                             EX POINT GD R
                     16 DIVE + 3 L        18 VEER + 7 L                             18 VEER + 3 R
                                          02 OPT QB + 1 L                           18 VEER + 3 R
                                          79 CURL INC L                             EX POINT GD R
                                          79 FLOOD INT L                            18 VEER + 6 R
                                                                                    EX POINT NG R
                                                                                    09 PITCH + 2 R
                                                                                    54 SP POWER + 2 R
                                                                                    07 POWER + 2 R
                                                                                    QB SNEAK + 1 R
                                                                                    EX POINT GD R
                                                                                    03 POWER + 4 R
                                                                                    03 POWER + 2 R
                                                                                    EX POINT GD R
                                                                                    02 OPT TB + 5 L
                                                                                    EX POINT NG L
                                                                                    03 POWER + 1 L
                                                                                    01 PITCH + 5 L
                                                                                    EX POINT NG L
                                                                                    QB SNEAK + 1 L
                                                                                    EX POINT GD L
```

Education (From Page 18)

Eighth is a "Series Chart"; in this report, information on each play (including the play name, number of yards gained or lost, initial of the scouted team's opponent for the scouted game, and vertical field position) is sorted by series number, to show how the scouted team is putting its drives together in different parts of the field. The ninth and tenth reports, the list of "All Plays That Gained 4 or More Yards" and the list of "All Touchdown Plays," highlight the most important plays of the game. Both of these reports include a printed diagram of each initial formation out of which the plays were run.

```
                              * ALL TOUCHDOWN PLAYS *
OPPONENT:

YEAR:

                                  O O O X O O O
                                        O
                                    O   O   O

                     T / NO ADJUSTMENT TO FORMATION / T

                                  QB SNEAK + 1 R

------------------------------------------------------------------------------------

                              * ALL TOUCHDOWN PLAYS *
OPPONENT:

YEAR:

                                  O O O X O O          O
                                        O            O
                                    O     O

          WIDE SLOT RIGHT OPEN - 3 / TAILBACK FLY (WEAK) / PRO LEFT OPEN
                                  18 VEER + 3 R

------------------------------------------------------------------------------------
```

(One segment is printed for each touchdown play)

When you use CHAMPS, you'll set up one distinct file, or collection of team data, for every team that you scouted. How many files you can fit on a single diskette depends upon the TRS-80 model that you have, and upon the amount of scouting data contained in each file. The Model III CHAMPS program diskette stores complete descriptions of 5500 plays, no matter how many files these plays are stored in, while the capacity of the Model I is 700 plays. If you have access to a computer system with more than one disk drive,you can use formatted diskettes to make your storage capacity as large as you wish. Otherwise, you can delete old files when they are no longer useful to make room for new entries.

Coaches Rauh and Dewey used CHAMPS in its various stages of development over the four-year period (beginning in 1976) that they were coaching together at Apple Valley. During that time, their team attained an overall 36-7 record, won the conference championship every year, and earned two state runner-up trophies. In the fifth year, the conference championship was won by Rosemount High School, where Rauh and Dewey are now administrators. In an eight-team conference, the championship team every year since CHAMPS was developed has been one of the two teams that had access to the system.

No scouting program can guarantee a team the kind of success that Apple Valley enjoyed, but Dewey and Rauh felt that as "a scouting system that could effectively and efficiently collect, analyze, and process data," CHAMPS "was a major factor" in their team's success. CHAMPS ™ : A Football Scouting Program will soon be available at your nearby Radio Shack Computer Center. CHAMPS ™ Richard Dewey and Mathew Rauh

French Verb Conjugation

John T. DeRegnaucourt Age 15 Centerville, Ohio

"This program entitled 'French Verb Conjugation' will conjugate any regular 'er', 'ir', or 're' verb. (It will not conjugate reflexive verbs.)

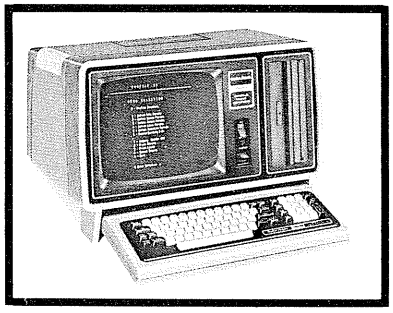
"When you run the program, input the infinitive form of the verb. It will be conjugated in three tenses: present, past and future.

"The French accents cause several problems since they cannot be input from the keyboard, so no accents can be input in the body of the verb; however, the accent aigu is poked (not printed) for the past tense 'er' suffix.

"This program was derived on a TRS-80 Model III computer. I do not know the required changes for making this program compatible with other RADIO SHACK computers, especially because of the poke statements."

```
10 ' By John T. DeRegnaucourt
20 ' FRENCH VERB CONJUGATION
30 CLS
   : CLEAR 500
   : DEFINT A-Z
40 PRINT TAB(20) "FRENCH VERB CONJUGATION"
   : PRINT
50 PRINT "  This program will conjugate any regular
   'er', 'ir', or 're'"
60 PRINT "verb.  Just input the infinitive form of the
   verb."
70 PRINT@320, "";
   : INPUT "THE VERB "; V$
80 FOR I=1 TO LEN(V$)
90 IF MID$(V$,I,1) < "A" OR MID$(V$,I,1) > "Z" THEN
   70
100 NEXT
   : IF LEN(V$) < 2 THEN 70
110 T$=RIGHT$(V$,2)
   : T1$=LEFT$(V$,1)
   : T2$=LEFT$(V$,LEN(V$)-2)
115 RESTORE
120 FOR I=1 TO 8
   : READ A$(I)
   : NEXT
   : FOR I=1 TO 8
   : READ B$(I)
   : NEXT
130 FOR I=1 TO 8
   : READ C$(I)
   : NEXT
   : FOR I=1 TO 8
   : READ D$(I)
   : NEXT
140 FOR I=1 TO 8
   : READ E$(I)
   : NEXT
   : FOR I=1 TO 8
   : READ F$(I)
   : NEXT
145 IF T1$="A" OR T1$="E" OR T1$="I" OR T1$="O" OR
   T1$="U" THEN A$="J'AI " ELSE A$="JE "
150 IF T$="ER" THEN 160 ELSE IF T$="IR" THEN 180
   ELSE IF T$="RE" THEN 200 ELSE 70
160 GOSUB 220
   : PRINT A$; T2$; A$(1)
170 FOR I=2 TO 8
   : PRINT D$(I); T2$; A$(I)
   : NEXT
   : GOTO 440
180 GOSUB 220
   : PRINT A$; T2$; B$(1)
190 FOR I=2 TO 8
   : PRINT D$(I); T2$; B$(I)
   : NEXT
   : GOTO 440
200 GOSUB 220
   : PRINT A$; T2$; C$(1)
210 FOR I=2 TO 8
   : PRINT D$(I); T2$; C$(I)
   : NEXT
   : GOTO 440
220 CLS
   : PRINT TAB(12) "THE PRESENT CONJUGATED FORMS OF "
   V$
230 PRINT
   : PRINT"PRESENT"
   : PRINT
   : RETURN
240 IF T$="ER" THEN 250 ELSE IF T$="IR" THEN 300
   ELSE IF T$="RE" THEN 320
```

(Continued on Page 29)



Model II

Product Line Manager's News

SCRIPSIT™

Many of you may be wondering what word processing is all about. To put it very simply, word processing (WP) is the automation of text handling from entry (when you type it) through the final copy (when it leaves your desk).

Let's follow a typical document from entry to final copy in two different offices.

Office One is a moderately successful small business with three employees. There is no secretary.

In this office, the boss types her own letters. Since she had typing courses in high school, this is no big problem. She usually types her letters as she thinks of what to say, so there are always several typed versions of her letters. This takes much longer than she would like, but she wants her letters to look as nice as possible (image is **everything** in a small business).

The re-typing is bad, but the worst problem is that no matter how hard she tries, she always ends up with a spot or two of whiteout on the final copy.

Office Two is a large corporation. They have 15 secretaries, each of whom is an excellent typist.

Here, letters and memos are hand written or dictated on cassette. They are then sent to the typing pool to be typed. The first draft is then sent back to the originator for verification and corrections. The corrected draft is then re-typed, and sent back to to the originator for signing. This cycle may occur several times.

Since the entire document must be typed two or more times before signing, the individual cost to produce each document is more than it could be. Also, as with Office One, final copies tend to end up with a spot or two of whiteout.

As you can see, both offices have the same problem. Productivity (or effort required to produce a document) **vs**. quality (how the final copy looks). It is obvious that an increase in productivity (fewer re-typings) will save money immediately. What is not quite so obvious is that this increase in productivity may well decrease quality of the final copy—which, in the long run, will damage the company's image (and thereby reduce sales).

Enter the word processor!

The function of every word processor is to get rid of the drudgery of re-typing entire documents. To do this, WP systems capture the information typed and save it in some form (such as a floppy diskette). The system then allows the typist to easily modify the saved text by adding, deleting, and overtyping while it handles all of the text shuffling. Thus, instead of retyping an entire letter just to change two words, the typist just corrects the words and asks the system to print the new version of the letter.

A word processor can provide both an increase in productivity and an improvement in quality at the same time. Since nothing ever gets on paper until the typist requests a printed copy, you should never see whiteout on a letter produced by a word processor. If the typist makes a mistake while typing, he or she can just backspace and then continue with the correct text. Mistakes are corrected on the video, not the paper.

Radio Shack's SCRIPSIT™ Word Processor!

Radio Shack has recently released SCRIPSIT 2.0. This program is a very sophisticated word processor. It allows a large range of flexibility and will handle most companies' (and individuals') requirements.

SCRIPSIT is a page oriented word processor. This means that text is stored and displayed in terms of a printed page of text. Also, SCRIPSIT is a 'what you see is what you get' system. This indicates that the text will print on paper almost exactly as you see it on the screen.

Working with SCRIPSIT is like working with an electric typewriter with a few important differences.

1. SCRIPSIT watches the line endings for you. When you approach the end of a line, SCRIPSIT automatically determines if the word will fit. If not, it moves that word to the next line. This means that you don't have to listen for the little bell and watch for the point that you want to end the line.
2. SCRIPSIT allows you to store 12 different sets of tab and margin settings. Just request your desired format by number.
3. SCRIPSIT will center lines of text automatically.
4. SCRIPSIT can underline and print text in bold. (Have you ever tried to make a phrase **stand out** by typing each letter several times with a typewriter??)

SCRIPSIT has a very large number of features and capabilities, but even a two finger typist (like me) can easily use it. The program is fully menu oriented and tells you where you are and what you can do next. Commands are consistent and easy to remember. Error messages are friendly and informative. In short, the system tries to be helpful, not intimidating.

The only keys that you need to remember to get started are **[F1]**, **[F2]**, and **[BREAK]**. The **[F1]** key tells SCRIPSIT to insert. When the directory of documents is on the screen, SCRIPSIT will insert a new document. If you are editing a document, SCRIPSIT will allow you to insert new text. The **[F2]** key is for delete. It allows you to delete characters, words, sentences, paragraphs, pages, and/or full documents depending on where you are at the time. The **[BREAK]** key is used to clear error messages and cancel commands.

Just knowing these three keys, **you** can use SCRIPSIT. As you become accustomed to the system, you may want to learn about **[ESC]**, **[TAB]**, and **[HOLD]**. But these are not necessary. The system was designed to be easy to learn, easy to use, and easy to add to. There are several add on features being considered, and one of them is available now!

SCRIPSIT Spelling and Hyphenation Dictionary!

This program will compare each word in your document to a 100,000 word dictionary. If the word is not found, the program then looks in a dictionary of up to 2047 words that you have added. If the word is still not found, you are given the choice of correcting it, adding it to your user dictionary, flagging it for later investigation, deleting it, or just skipping it. The speed of this program averages about 2000 words per minute.

Once you have verified the spelling, you can request hyphenation. The program will pick the most appropriate point for syllabication from the selection stored in the dictionary. This process is slightly slower than spelling verification, but the result is worth the wait.

The SCRIPSIT Spelling and Hyphenation Dictionary is the most accurate system (compare us to IBM) available on the market today!

(Continued on Page 21)

Model II (From Page 20)

In Closing

There isn't enough space available here to list all of SCRIPSIT's features and capabilities. But if you will drop by your local Radio Shack Computer Center and take a look at SCRIPSIT, they will give you a demonstration. I think you'll be glad you went.

ANSWERS

Bruce W. Elliott

We recently received two letters dealing with Profile II and/or Model II Scripsit. In both letters were questions about how to do something, or suggestions for product improvements. We would like to reprint portions of these letters, and try to provide some answers. One of the letters was from John O'Rourke of the Southwest Law Office in Chicago Illinois, and the second was from John Hawkins Vice President of K-Fox Radio Station in Seattle, Washington.

Since both letters dealt at least partially with Model II Scripsit, we will look at Scripsit first.

Both users seemed to be having problems with creating lines, or with underlines. The Southwest Law Office wanted us to "get rid of the clumsy codes for making lines (I waste more paper because of an omitted or an extra underline code)." K-Fox had this problem: " . . . we can't make the DWII print a line. It will underline, but it won't just print a line like a typewriter will. . ."

Using Scripsit 1.0, underlining is very straight forward. Surround the text you want underlined with single underline characters. The underline character is **[SHIFT][HYPHEN]**, the first one encountered turns the underline function on, and the second one turns it off. From the manual (page 55): "The underline code is treated as one space when the document is printed. However, if one (or both) of the characters that surround an underline code is not a letter or a number, then the underline character will not leave a space (when the document is printed)."

Where problems can occur, is when you want just a line, not an underline. What we really want here is not an underline (since we are not drawing a line under anything), but to print a series of underscores. Page 54 of the Scripsit Reference Manual details several "Swap Codes." A swap code simply tells Scripsit that you are redefining a particular code. To draw a line with Scripsit 1.0, you must first use the swap code **[CTRL][9][6][2]**. What this does is to tell Scripsit to print an underscore character wherever it finds the underscore (Shift Hyphen) character, rather than turn underlining on or off. If you always want lines, rather than underlining, you can place this control sequence at the very beginning of your document, then simply use the Shift-Hyphen character wherever you want underscores in your document.

To turn the underscore feature off and return to normal underlining, repeat the control sequence **[CTRL][9][6][2]**.

What the manual fails to tell you is that the control sequence needs to be on a line all by itself. Here is the sequence to use to draw just a line using Scripsit 1.0:

[CTRL][9][6][2][ENTER]
[ENTER]
[CTRL][9][6][2][ENTER]

The only thing that is "critical" is that the control sequences each be on a line by themselves.

By using appropriate combinations of **[CTRL][9][6][2]** and surrounding **[SHIFT][HYPHEN]**s, you should be able to combine underscores and underlines on the same page with very little trouble.

Scripsit 2.0 has simplified these procedures considerably. To underline text, you now surround it with **[ESC][SHIFT][HYPHEN]** characters (note: one additional difference is that you will need to put in space characters if you want spaces). The **[ESC][SHIFT][HYPHEN]** prints an extra wide underscore character on the screen so it is easy to spot. To draw a line using underscores, simply draw the line you want with underscore characters (**[SHIFT][HYPHEN]**), Scripsit 2.0 will do the rest. (How's that for getting rid of clumsy codes?)

Mr. O'Rourke also wanted to know what could be done about the clumsy codes for special printer characters like ½, accents, umlauts, etc. Well, Scripsit 2.0 gives you good things and bad. First the bad. We were not able to completely eliminate the control codes. The good news is that we gave you easy access to those special characters, and shortened the control codes on the display (two reverse video characters instead of the string of several characters with 1.0).

In Scripsit 1.0, if you wanted to access the TM character on the Daisy Wheel II printwheel, you had to PATCH Scripsit, and then use a **[CTRL][6][9][n]** control sequence (where n was the number of one of the ten special codes you could access). With Scripsit 2.0, the main utilities menu has a special feature which allows you to define up to 26 special printer codes. You can use these codes for special characters like TM, or to issue special commands to the printer, like enter 12 pitch or proportional space modes.

Each of the special sequences that you set up is assigned to a letter in the alphabet. When you want to use that code, press **[CTRL][X][k]**, where k is the letter of the alphabet you assigned the function to. I assigned TM to the T key. When I need a TM, I use **[CTRL][X][T]**. On the video I see a lower case x and an upper case T, both in reverse video. I find it a lot easier to remember that xT is TM on my Scripsit 2.0, than to remember that ~ \ 1 was TM when I was using Scripsit 1.0. Hopefully this will help with one of Mr. O'Rourke's other problems, which was the difficulty of proofreading material with all of those weird 1.0 control sequences.

Something that should be mentioned about these new easy to define printer controls is that you are not limited to a single character like you were in Scripsit 1.0. I have frequent need for a slashed zero. In Scripsit 1.0 I had to use 0~ \ / , while now I use **[CTRL][X][Z]**. On the video all I see is xZ, but I know that it will print a slashed zero. Your printer control sequences can be up to 8 bytes long, which gives you a lot of control over the printer.

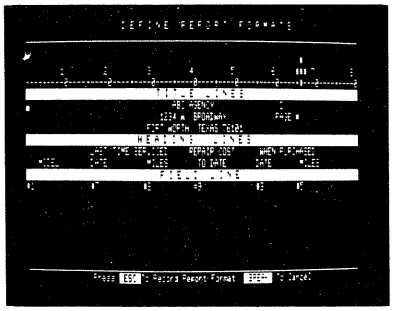
Mr. O'Rourke also suggested that it is very tedious to have to enter information between braces when you are trying to set up a Scripsit merge file. We agree. Here is what we think is a "better" way. First, type all of the information without braces. Once you have finished typing the information, let the computer insert the braces for you. If you are using Scripsit 1.0 or 2.0, you can define a single key that will place the braces around your information. Lets say you have a NAME, ADDRESS, CITY, STATE, and ZIP that you want to merge.

In Scripsit 1.0, go to the main menu, select Utilities (U). Under Utilities, select Change Disk Defaults (C) and User Key (U). I chose to set this up on the K key:

```
HOLD LEFTARROW F1 { F2 HOLD RIGHTARROW F1 } F2
ENTER
F1 { F2 HOLD RIGHTARROW F1 } F2 ENTER
F1 { F2 HOLD RIGHTARROW F1 } F2 ENTER
F1 { F2 HOLD RIGHTARROW F1 } F2 ENTER
F1 { F2 HOLD RIGHTARROW F1 } F2 ENTER
```

With the K user key set up this way, simply press **[CTRL][K]** at the beginning of each set of names and addresses. Scripsit will automatically place the braces around the text for you.

(Continued on Page 23)



Profile

SMALL COMPUTER COMPANY

Written for the users of PROFILE II and PROFILE +
PROFILE II Editor
P.O. Box 2910
Fort Worth, Texas 76113-2910

MAY WE INTRODUCE OURSELVES?

Beginning in this issue, we'll be writing a monthly column in the Radio Shack Newsletter. We plan to present new applications, modifications (some from you, and some from ourselves), and answers to queries from PROFILE II and PROFILE + users. Your feedback is our most important asset: we welcome both questions and suggestions, and in future issues we will reprint as many as we have space for. Please write to us at the address at the top of this page.

PREPARING TO CREATE A PROFILE II DATABASE

Mr. Larry Small of Ft. Wayne, Indiana, writes: "Dear PROFILE II Column: I often find myself having to create a PROFILE II database several times in order to get it right. Do you have some suggestions on some steps I might take before taking key to keyboard?"

Probably the most important "step" to take before creating a PROFILE II database is to plan. Too many users begin by sitting down and defining a file format only to find out later (perhaps many keypunching hours later) that they were missing a field or two. Well-defined databases are a successful blend of several concepts; (1) flexibility, (2) maximized storage, and (3) usefulness. Such blending comes only from careful planning.

Flexibility is the ability of a database to meet not only your present needs, but your future ones. Do you have a plan for using the 9 or 10 digit zip code in the future? Might there be additional fields that you will want to select or sort by? Asking yourself questions like this will help insure that you will not make your database obsolete.

At the same time you try to make your database flexible, make it storage-efficient. Sure, putting four unused ten-character fields into your first segment will help you deal with future needs, but is it worth the storage it will take up?

Third, the database must be useful. The fields that you intend to sort on must be in the first segment. The data entry screens must be clean and easy to understand. Remember, you bought a computer to help you, not to create extra work.

Perhaps the best way to discuss the proper method for setting up a PROFILE II file is to use an example. In this example, you're going to store the membership list of the Detroit chapter of the Friends of the Wolverine Society. Step one is to make a list of the information that you wish to hold on each member. Your list might look like this:

First name (10 chars)	Last name (15)
Employer's name (20)	Spouse's name (10)
Street address (30)	City (15)
State (2)	Zip code (5)
Home phone (8)	Office phone (8)
Status (2)	Date membership (8)
Date Renewal (8)	Membership type (2)
Newsletter (Y/N) (1)	Child 1's name (10)
Child 2's name (10)	Child 3's name (10)

Many users might stop at this point and begin defining the database. It is important, however, to arrange this information in a better fashion.

The next step is to go over the list and flag the items that you wish to search and sort by (first segment items):

*First name (10)	*Last name (15)
*Employer's name (20)	Spouse's name (10)
Street address (30)	City (15)
*State (2)	*Zip code (5)
Home phone (8)	Office phone (8)
*Status (2)	*Date membership (8)
*Date Renewal (8)	*Membership type (2)
*Newsletter (Y/N) (1)	Child 1's name (10)
Child 2's name (10)	Child 3's name (10)

Let's discuss some of these choices. Although you rarely scan by first name (e.g., show all members named "Larry"), it is nice to have an alphabetical list that sorts by first name within last name. (This will insure that Bob Smith is earlier on the list than Zolton Smith.) The street address does not belong in the first segment! Street addresses are so varied that sorting by them will do nothing but separate those who use post office boxes. Selecting by street address is made nearly impossible by the many ways of spelling an address. (For example: 151 Seventh Ave., One-Fifty-One 7th Avenue, 151-7 Ave., etc.). Remember, it's nice to have everything in the first segment, but you want to have room for future expansion.

Now that you've isolated the items that belong in the first segment, let's list them and find out how much room you've used. The fields are:

*First name	10
*Last name	15
*Employer's name	20
*State	2
*Zip code	5
*Status	2
*Date membership	8
*Date Renewal	8
*Membership type	2
*Newsletter (Y/N)	1
	73 characters

Not a bad first segment. You have twelve characters for future expansion. Now—should you leave the segment in this order? The answer is no. Remember that PROFILE fields that are consecutive (next to each other) can be sorted together: the first field will be the major sort and the next field will be the minor sort. Therefore, you want First name to FOLLOW Last name. What if you have large numbers of members who work for a few key employers? Perhaps you would like to produce a listing grouped by employer, sorted by individual within each employer. This would mean that Employer's name must precede Last and First names.

Let's now turn our attention to the data segments. Here are the fields that remain:

Spouse's name	10
Street address	30
City	15
Home phone	8
Office phone	8
Child 1's name	10
Child 2's name	10
Child 3's name	10
	101 characters

(Continued on Page 23)

Profile (From Page 22)

If you will eventually be keeping thousands of records in the file, you will want to save disk space by using variable length segments for these fields. (If you are not familiar with variable length segments, you may wish to refer to the Microcomputer Newsletter of June, 1981. PROFILE + users can consult their manual.)

The order of the remaining fields is not important, but you may wish to divide them into two data segments. Because the data in each segment can reside on a separate disk, you will be able to spread the data fields over two drives. This is not something that has to be done right away, but having it in two segments will give us that opportunity.

Therefore, the remaining fields will be put into Segments 2 and 3. For future expansion, you'll want to make Segment 3 a little larger than needed. A good plan might be to make Segment 2 fifty characters and Segment 3 seventy characters. This will permit you to add fields without going to a fourth segment.

As you can see, a bit of planning can help to make a PROFILE database much more usable. Learning to use a system like this is a little like learning to drive. There's no substitute for experience! Before you spend valuable time on your main application, experiment! Happy filing!

Hope you like this sample of what we intend to offer to you in the future; please write us at the address at the head of this column and let us know.

See you next month—

Answers (From Page 21)

For Scripsit 2.0 you can set up the same function while you are in your document, no need to return to the main menu (just use one of the 2.0 control keys).

Mr. O'Rourke also asked for several features to improve editing. These changes included the ability to highlight frequently changed portions of text, make portions of the text unchangeable, edit merge documents after the merge information was in place and the ability to merge from more than one file. Sorry, some of these sound interesting, but are just not possible.

Mr. Hawkins had a problem with printing leading blanks on the Daisy Wheel II. Normal space characters, created by pressing the space bar, are "soft" spaces. When a soft space occurs at the beginning of a line Scripsit will ignore the space. This is done to allow proper left justification of your documents. If you need to insert spaces at the beginning of a line, use "hard" spaces. A hard space is created using the **[F1]** key. The small box-like dots that the **[F1]** key generates can be left at the beginning of a line to force Scripsit to insert spaces at the beginning of a line. This information is covered in the manual, but may be "hidden." I found it in the discussion on inserting text, Page 35 of the manual.

Check with your local Radio Shack store about an upgrade from Model II Scripsit 1.0 to Scripsit 2.0 which should be available by the time you read this.

Mr. Hawkins had several questions or problems with Profile II. The first was that he needed to randomly select records and have them printed. The problem was that if he used the screen print function, each record takes an entire page, which can be quite wasteful. The Report utility did not help because there was no way to indicate which records were to be printed.

Our chief Profile person suggests that you create an additional field in segement 1 (he suggested a single byte, I would use 4 bytes) which could be used to flag the entries you want. Then when you want a particular record to be printed, place a "flag" or character into the flag field. When you run the Report utility, sort for records with the correct character in the flag field. I would use a 4 byte field so the flag could be day and month. That way, I simply enter the day and month in those items I want flagged. When all the records have been selected, exit to the report program and sort on the day and month flag field. The advantage is that there is little likelihood (at least for a year) that I will use the same flag code twice, and that means I do not have to be as careful about erasing all of the old flags. It would also give me information about the last time a particular record was accessed.

Mr. Hawkins correctly pointed out that the hard copy function in Add/Update is simply a screen print function. What you see is all that will be printed.

Another of Mr. Hawkins' questions concerned interfacing Profile II data and a BASIC program. This is quite easy to do, and is covered in the Profile II article in the November issue. We would also refer you to the SELECT/BAS program which comes with Profile II. Studying this program should give you valuable information about how to read and update Profile data files from your BASIC programs.

I hope this information has been useful to both Mr. Hawkins and Mr. O'Rourke, as well as those of you who took the time to read it.

Notes on Previous Newsletters

June, 1981

Reduce Fractions

From Mr. Richard Shepherd Chalmette, La.

I read your June issue of Newsletter, and I noticed that one program is wrong. Sheldon Beasley's program for reducing fractions contains a few errors. Below is a slightly modified program which I made.

I changed Sheldon's line 50 which read:

```
5Ø  FOR X= A TO 1 STEP -1
```

to my line 30 which reads:

```
3Ø  FOR X= A TO 2 STEP -1
```

I also inserted line 20.

```
5 CLS
1Ø INPUT "INPUT FRACTION (A/B) AS A,B"; A, B
2Ø IF A=1 GOTO 8Ø
3Ø FOR X=A TO 2 STEP -1
4Ø IF A/X = INT(A/X) GOTO 6Ø
5Ø NEXT X
    : GOTO 8Ø
6Ø IF B/X = INT(B/X) GOTO 9Ø
7Ø GOTO 5Ø
8Ø PRINT "IN LOWEST TERMS"
    : GOTO 1ØØ
9Ø PRINT A; "/"; B; "CAN BE REDUCED TO"; A/X "/"; B/X
1ØØ PRINT "FOR ANOTHER FRACTION TYPE"
11Ø INPUT "1, ELSE TYPE Ø"; Q
12Ø ON Q+1 GOTO 13Ø, 5
13Ø END
```

From Dr. Robert F. Cathcart, III San Mateo, Ca.

The program to REDUCE FRACTIONS in the June Newsletter contains an error that makes reductions of fractions such as $^{10}/_{30}$ impossible. Enclosed is a correction. Also enclosed is a more complicated program which will reduce compound fractions and give a fraction equivalent. The fraction is entered as with $^{24}/_{28}$ or $5^{16}/_{20}$. It will reduce fractions as 567 $^{344}/_{24}$ but needs a little time.

HERE IS THE CORRECTED VERSION:

```
2 REM SHELDON BEASLEY
4 REM ATLANTA GEORGIA
6 REM MODIFIED BY ROBERT CATHCART
8 REM SAN MATEO CALIFORNIA
1Ø CLS
2Ø PRINT "INPUT FRACTION (A/B),"
3Ø INPUT "AS A,B"; N, D
35 IF N/D=1 THEN 1Ø5
36 IF N/D<1 THEN 4Ø
```

(Continued on Page 24)

Notes (From Page 23)

```
37 B=N
     : C=D
     : GOTO 5Ø
4Ø B=D
     : C=N
5Ø FOR X=N TO 1 STEP -1
6Ø  IF INT(N/X)-N/X=Ø THEN 9Ø
7Ø  NEXT X
8Ø GOTO 13Ø
9Ø IF INT(D/X)-D/X=Ø THEN 11Ø
1ØØ GOTO 7Ø
1Ø5 PRINT "FRACTION EQUALS 1"
     : GOTO 13Ø
11Ø IF X=B THEN PRINT "IN LOWEST TERMS"
     : GOTO 13Ø
12Ø PRINT N/X; "/"; D/X; "IS THE REDUCED FORM OF "; N;
    "/"; D
13Ø PRINT "FOR ANOTHER FRACTION TYPE"
14Ø INPUT" 1, ELSE TYPE Ø"; Q
15Ø ON Q+1 GOTO 16Ø, 1Ø
16Ø END
```

HERE IS MY MORE ADVANCED VERSION:

```
2 REM ROBERT CATHCART
4 REM SAN MATEO CALIFORNIA
1Ø CLS
15 PRINT
2Ø PRINT "ENTER FRACTION (N/D) OR "
25 INPUT "ENTER COMPOUND FRACTION (I N/D)"; F$
26 F1$=F$
3Ø GOSUB 1ØØØ
35 IF N/D>1 THEN 4Ø
37 B=D
     : GOTO 5Ø
4Ø B=N
5Ø FOR X=N TO 1 STEP -1
6Ø  IF INT(N/X)-N/X=Ø THEN 9Ø
7Ø  NEXT X
8Ø GOTO 15
9Ø IF INT(D/X)-D/X=Ø THEN 11Ø
1ØØ GOTO 7Ø
11Ø IF N/D<1 AND N=N/X AND D=D/X THEN 2ØØ
12Ø IF D/X = 1 THEN 21Ø
14Ø IF N/D > 1 THEN 22Ø ELSE 25Ø
2ØØ PRINT F1$; "IS IN LOWEST TERMS"; "="; N/D
     : GOTO 15
21Ø PRINT F1$; " ="; N/X
     : GOTO 15
22Ø R = N/D - INT(N/D)
     : W = R*(D/X)
23Ø PRINT F1$; " ="; N; "/"; D; "=";
24Ø PRINT INT(N/D); INT(W+.1); "/"; D/X; "="; N/D
     : GOTO 15
25Ø PRINT N; "/"; D; "="; N/X; "/"; D/X; "="; N/D
     : GOTO 15
1ØØØ F = LEN(F$)
1Ø2Ø FOR Y = 1 TO F
1Ø25 IF MID$(F$,Y,1) = " " THEN GOTO 2ØØØ
1Ø3Ø IF MID$(F$,Y,1) = "/" THEN GOTO 1Ø5Ø
1Ø4Ø NEXT
1Ø5Ø N$ = LEFT$(F$,Y-1)
1Ø6Ø D$ = RIGHT$(F$,F-Y)
1Ø7Ø N = VAL(N$)
     : D = VAL(D$)
1Ø71 IF C = Ø THEN 1Ø8Ø .
1Ø72 CF = C * D
1Ø73 N = N + CF
1Ø74 C = Ø
1Ø8Ø RETURN
2ØØØ C$ = LEFT$(F$,Y-1)
2Ø2Ø C = VAL(C$)
2Ø3Ø F$ = RIGHT$(F$,F-Y)
2Ø4Ø GOTO 1ØØØ
```

Get Your DATA for Free

From Mr. John Muczynski ST. Clair Shores, Mi.

In the June issue of your Newsletter there is an article written by William Terrel called "Get Your Data For Free." (pp. 8-10).

You noted that after the program 'AUDATA' converted the data 1, 2, 3, . . . 10 into BASIC DATA lines it was in memory as 254, 253, 252, . . . and was converted wrong on the second run. This problem can be fixed by adding the following line to the source code:

```
765  LD   (IY),A     ;PUT BACK ORIG. BYTE
```

Also, another way of obtaining a starting line for the data is by starting at the thousand after the last line number in the resident BASIC program. For example:

If the last line is

```
6350 END
```

then the DATA lines would start at 7010 (the 10 for a byte counter as pointed out in the article).

Guess Number

From Donald Parson, of DelRay Beach, Florida, comes this note:

Regarding Les Childers' "Guess Number" program in the June "Microcomputer News," why limit the entering number to 1000? It works fine with larger numbers. Then, line 30 could be deleted and line 10 rewritten:

```
1Ø:PRINT "ENTER NUMBER"
```

I have taken the liberty of modifying the already clever program, thus making it easier and simpler to use:

```
1Ø:CLEAR
2Ø:INPUT "ENTER NUMBER   ",X
4Ø:X=INT(X*X/12+1Ø-3Ø)
5Ø:INPUT "YOUR GUESS IS   ";Y
6Ø:IF Y=X THEN 9Ø
7Ø:IF Y>X THEN 1ØØ
8Ø:PAUSE "YOUR GUESS IS LOW"
    : GOTO 5Ø
9Ø:BEEP 5
    : PRINT "YOUR GUESS IS CORRECT"
    : GOTO 1Ø
1ØØ:PAUSE "YOUR GUESS IS HIGH"
    : GOTO 5Ø
11Ø:END
```

July, 1981

LISTER/BAS

From Mr. William Cornwell Cary, North Carolina

The concept of the LISTER/BAS program (published on page 7, July 1981 issue) is tremendous. As published the program will not run!

Corrections I made include:

```
55 OPEN "I",1,F$
```

(In order to read a disk file, it first must be open)

```
9Ø IF D<48 OR D>57 THEN PRINT "* NOT ASCII BASIC FILE *"
     : CLOSE
     : GOTO5Ø
```

(File opened in statement 55 must be closed—even if file in error)

```
172 IF SP>2 OR SP<1 THEN 17Ø
```

(Verify that a 1 or a 2 indeed was received—minor correction)

Another change that could have been included is:

* Printing "END OF LISTING" at the end of the listing for positive feedback that the listing is complete.

LISTER/BAS is already one of my most popular utility programs for developing programs in BASIC. Thanks for the motivation and concept. (Continued on Page 25)

Notes (From Page 24)

Editors Note: Mr. Cornwell was one of several people to tell us of our omission of an OPEN statement. As we noted in the September Newsletter, the OPEN statement was intended to be at the end of line 60. We would like to thank those of you who called this error to our attention.

From W. L. Garrott we got a copy of a letter which he sent to Ed Faulk, Editor of OCTUG. Part of that letter contained this information about the LISTER/BAS program:

"The July issue of RS's Microcomputer News has a reprint of their LISTER program. It is my most useful one. They omitted the file opening in line 60. I have made changes to provide a left margin (I like to file programs in 3 ring notebooks) and to place the page number with respect to the end of the lines . . .

"The changes are:

```
6Ø ON ERROR GOTO 4ØØØ
     : OPEN "I", 1, F$
26Ø W1=WD
     : T=1Ø
61Ø LPRINT TAB(1Ø); V$;
     : IF L>=WD THEN LPRINT"  "
     : GOSUB 7ØØ
     : LPRINT TAB(1Ø);
     : L=1Ø
64Ø IF V$=":" AND NN<>1 THEN LPRINT"  "
     : GOSUB 7ØØ
     : LPRINT TAB(T+5);
     : L=5
2ØØØ PG=PG+1
     : LPRINT TAB(1Ø); LEFT$(TL$,3Ø);
     : LPRINT TAB(WD-12); DT$;
     : LPRINT TAB(WD-8); USING"        PAGE ##"; PG
2Ø1Ø LPRINT TAB(1Ø); STRING$(WD,"=")
```

"The space following USING" is necessary because of the maximum tab of 63 for the Model I."

August, 1981

Checker Board

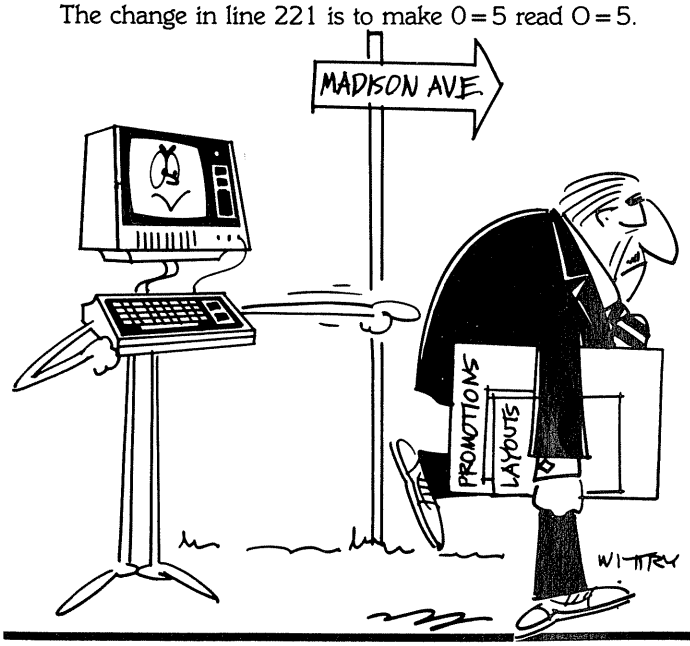
Our Quality Control department (QC) pointed out three typographical errors (ours) in Mr. Cotton's Checker Board program on page 12 of the August 1981 issue.

In lines 137 and 138 change the word PRINT to PAINT.

Line 221 should read:

```
221 IF O=7 THEN 224 ELSE IF O=5 THEN 224 ELSE IF O=3 THEN
    224 ELSE IF O=1 THEN 224
```

The change in line 221 is to make O = 5 read O = 5.



Dow Jones (From Page 6)

"George! You and DJ are terrific." I disconnected from DJ with a fond farewell (by typing DISC and **[ENTER]** key), and reached for the phone to call my husband at the office.

"Phil, we've struck oil. We're rich!"

There was silence on the other end so I told him about our stock.

"How did you know?" he questioned. "There wasn't anything in the papers this morning about it."

"It's a recent development" I said, trying to compose myself. "I just found out through our personal newspaper, Dow Jones Information Services."It was Phil's turn to be mystified, and I decided to wait until he got home to show off our new friend.

"By the way," he asked, "is George OK?"

"Oh yes. We're having a wonderful time—just the three of us. See you later Phil."And I hung up.

I was hooked on DJIS, and wanted to know more. In a few days I received information on other areas I could access, as well as the pricing, and updates on new services. Phil and I spent that evening with our children exploring data bases—somewhat like early discoverers locating new lands. Phil was fascinated with the financial statistics from Media General, including revenue, earnings, dividends, and other detailed information on 3200 companies and 180 industries, and the weekly economic survey and forecast from Money Market Services. As a result of this careful scrutiny, he decided to sell one of our stocks and invest in a few that looked promising. Of course he checked up on them first through their News stories and Current and Historic quotes data.

Our son mastered the Free Text Search data base and was able to research reports for school quickly and in depth by using key word searches. My favorite is still News, though I enjoy reviewing the Wall Street Week transcripts, on line from the Public Broadcasting television show.

Phil was sure that our DJ connection would cost us a fortune, but it has proved to be a wise investment. DJIS provides us with the reliable and timely information we need to buy and sell our stocks, and our profits have already paid for George. Our program and usage cost also may be tax-deductible (consult your tax advisor for further information). Also, because we can select just the information we want, our use of time is efficient, and we keep our costs low. Before I log on I know what categories or companies I want to follow, and I get the information quickly. Of course, if I can wait until evening, the prices drop much lower.

And DJ is always working on adding new information and data base enhancements for me. In fact I called the DJ Hotline a few days ago and discovered some new additions to the Service: the Dow Jones Weekly Economic Update, and The Corporate Earnings Estimator.I know Phil will like the Corporate Earnings Estimator. It will give him the consensus forecast of earnings prospects for 2200 major companies, provided by research analysts at 45 major brokerage firms. That seems like pretty good advice to consider when choosing our stocks. And the Weekly Economic Update allows us to find a week's worth of economic statistics in one place—including interest rate trends, consumer price index (helpful in family budget planning), producer price index, federal budget and tax issues (a favorite with Phil), and many other key indicators.

DJ is even providing a 1/2 hour of free time on each of these new services to try them out. Which is what I'm going to do right now. If you want to join me check DJHOW (.DJHOW 01) for all the details, or call that toll-free number for more information: 1-800-257-5114 (609-452-1511 in New Jersey).

PS. I may have doubted George in the beginning, but he's brought Dow Jones Information Services into our lives, and we're all one big happy family now. And I bet by next month there will be more exciting things to tell you about, and some helpful hints as I learn more and more about Dow Jones Information Services.

CompuServe (From Page 5)

The first phase includes CompuServe's news and information network. The regular "shopping list" of features available to CompuServe customers is augmented by United American Bank's own news and daily information on savings and deposit rates.

The second phase will allow customers to pay most of their regular monthly bills by displaying that information on their television screen; customers simply choose who to pay, how much and when. And customers will be in direct communication with the bank, so their bills can be paid on the precise day requested.

The third phase, and one which bank officials report has been in demand for some time, will allow customers to see their checking transactions on the television screen. A current list of all checks and deposits which have reached the bank that day and a running balance of the account will be provided. Sudman calls the feature "a great plus for in-home money managers," because the checking account capability enables customers to have instant awareness of the status of their accounts.

By the end of 1981, Sudman says, the Bank-at-Home service will also include bookkeeping programs, loan preparation and electronic mail services.

New Services Featured in Publications to Customers

CompuServe Information Service customers now receive two free publications to keep them informed on new services and features.

"Update" is a newsletter published monthly which tells customers about new products, new programs, telephone number additions and corrections and other information. Update will also carry the complete CompuServe Information Service index for you to keep as a reference.

"Today" is a color magazine which talks in depth about new services, information providers to CompuServe, customers and how they use the service and special announcements. Today is free to CompuServe customers and is available by subscription to noncustomers.

CompuServe is Available in Radio Shack Stores

Remember that the CompuServe Information Service is available in Radio Shack stores across the country.

Visit the store nearest you for a live demonstration of the service.

Questions and comments about CompuServe can be sent to Richard A. Baker, Editorial Director, CompuServe Information Service, 5000 Arlington Centre Blvd., Columbus, Ohio 43220 or through Feedback, main menu item 5, CompuServe User Information.

VisiCalc (From Page 8)

```
FØ8Ø   ØØ 7E 47 1A B8 3Ø Ø3 ØE   Ø1 47 AF BØ 28 19 C5 13
F Ø9Ø   23 4E 23 46 C5 E1 EB 4E   23 46 C5 E1 C1 1A 96 38
FØAØ   ØA 2Ø 53 13 23 1Ø F6 CB   41 2Ø 4B ED 4B 1D F1 AF
FØBØ   E1 ED 42 EB E1 ED 42 3A   1B F1 47 C5 18 ØD C5 ED
FØCØ   4B 1F F1 1B 1B 2B 2B Ø9   EB Ø9 EB 4E EB 7E 71 EB
FØDØ   77 23 13 4E EB 7E 71 EB   77 23 13 4E EB 7E 71 EB
FØEØ   77 C1 1Ø DA 2A 17 F1 EB   2A ØF F1 AF ED 52 22 ØF
FØFØ   F1 D2 58 FØ 18 Ø2 D1 E1   2A 11 F1 11 Ø1 ØØ AF 19

F1ØØ   22 11 F1 ED 5B 13 F1 ED   52 DA 52 FØ C3 3Ø FØ FF
F11Ø   FF FF FF FF FF FF FF FF   FF FF FF FF FF FF FF FF
F12Ø   FF 21 ØØ FØ E5 2A Ø3 28   E9 FE 4F 28 Ø2 CB C8 DD
F13Ø   7E Ø4 FF FF FF FF FF FF   FF FF FF FF FF FF FF FF
```

Listing 3.

Use this command line to DUMP a copy of the machine code to disk with the file name "SORTN":

```
DUMP SORTN START=FØØØ, END=F14Ø, TRA=F121, RORT=R
```

Electronic Flashcards

Nick Kasoff Yellow Spring, Ohio

The following compact two-liner program is one that any high school student struggling his or her way through Spanish, French, Latin, or any other language will greatly appreciate. When that big vocabulary test comes up, just whip out the TRS-80, and you have any easy 100% on it.

Note: This program accepts that data Spanish first, then English (the way most textbooks and dictionaries list it), and drills you in translation from English to Spanish (the way most teachers test you). In order to change this, make appropriate modifications in the PRINT or INPUT statements in line 10. Happy studying!!!

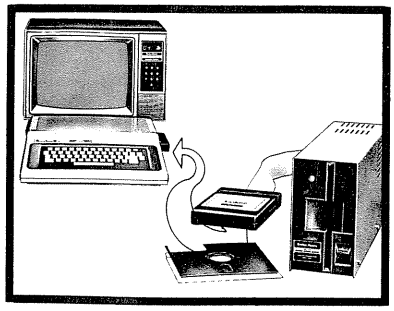
```
1Ø  CLS
    : CLEAR 1E4
    : DEFINT A-B
    : INPUT "NUMBER OF ENTRIES"; A
    : DIM A$(1,A)
    : PRINT "SPANISH, ENGLISH"
    : FOR B = 1 TO A
    : INPUT A$(Ø,B), A$(1,B)
    : NEXT
2Ø  B = RND(A)
    : CLS
    : PRINT A$(1,B)
    : INPUT A$
    : IF A$ = A$(Ø,B) THEN 2Ø ELSE PRINT "WRONG--- "
    A$(0,A) " IS RIGHT."
    : FOR N=1 TO 5ØØ
    : NEXT N
    : GOTO 2Ø
```

Subroutines (From Page 4)

The second routine is called "BLINK/BAS", and can be used to alert an operator that the computer is waiting for an input. The routine does this by blinking an underscore on and off at the current cursor position. The routine can be used in place of any single value input statement by replacing the input statement with GOSUB 60100. The subroutine will return the operator's response as a string variable (ZZ$) which can be converted to a numerical value by using the VAL statement as required.

BLINK/BAS

```
6Ø1ØØ REM BOB CROCKETT APRIL, 1981
6Ø11Ø ZZ$=""
6Ø12Ø Z%= 256 * (PEEK(16417)-6Ø) + PEEK(16416)
6Ø13Ø PRINT@Z%, CHR$(95);
6Ø14Ø FOR ZZ%= 1 TO 2Ø
6Ø15Ø Z$=INKEY$
6Ø16Ø IF Z$=CHR$(13) THEN PRINT@Z%, " ";
    : RETURN
6Ø17Ø IF Z$<>"" THEN 6Ø26Ø
6Ø18Ø NEXT ZZ%
6Ø19Ø PRINT@Z%, CHR$(32);
6Ø2ØØ FOR ZZ%=1 TO 35
6Ø21Ø Z$=INKEY$
6Ø22Ø IF Z$=CHR$(13) THEN RETURN
6Ø23Ø IF Z$<>"" THEN 6Ø26Ø
6Ø24Ø NEXT ZZ%
6Ø25Ø GOTO 6Ø13Ø
6Ø26Ø IF ASC$(Z$)<>8 THEN 6Ø31Ø
6Ø27Ø PRINT@Z%, " ";
6Ø28Ø PRINT@Z%-1, "";
6Ø29Ø ZZ$=LEFT$(ZZ$, LEN(ZZ$)-1)
6Ø3ØØ GOTO 6Ø12Ø
6Ø31Ø IF ASC(Z$)<32 THEN POKE 15360+Z%, 32
    : GOTO 6Ø12Ø
6Ø32Ø PRINT@Z%, Z$;
6Ø33Ø ZZ$=ZZ$+Z$
6Ø34Ø GOTO 6Ø12Ø
```



Color Computer

Product Line Manager's News

Well, here it is October. By now, I hope you have had a chance to pick up one of the new Radio Shack 1982 catalogs at your local "Shack" or Radio Shack Computer Center.

Over the past few months, I've been receiving lots of phone calls and letters telling me what the new Color Disk System will and will not be capable of. Well, maybe now I can set at least part of the record straight.

First, I would like to cover the concept of what we tried to do in developing the operating system for the disk system. Since our Models I, II, and III have operating systems which when used, generally require two stages of operation (TRSDOS READY and BASIC READY), we thought that for a home computer the operating system should be transparent to the user. This means that when the machine is turned on, the screen would appear just as if there were no additional operating system. If you wanted to run a BASIC program, you would not be required to load BASIC from TRSDOS, simply load the program and run. If, in the middle of the program, you "BREAK" and want to check the DIRectory of one of your diskettes, you can do so without destroying your program.

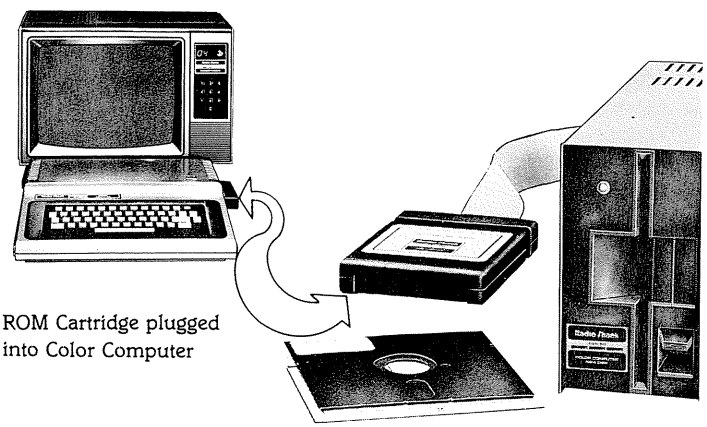
Additionally, we did not want to use a large amount of RAM memory (user's memory) for storing either the operating system or extensions to BASIC. Along the same path, we did not want to use storage capacity on the diskette for operating system utilities or instructions.

The design called for similar Input/Output instructions that are available on our other models allowing both direct and sequential access to data files on the diskette.

In a nutshell, we wanted a system which offered full disk routines/access but did not require an overlay of BASIC, simply an extention of existing BASIC which was already in ROM.

This is what we came up with (some of this information is in the Radio Shack catalog, so bear with me):

The "operating system" is contained in a ROM cartridge which simply plugs into the Program Pak slot on the side of the computer. It requires Extended BASIC (which requires 16K RAM memory). The complete Standard Color BASIC language and Extended Color BASIC language are left unchanged. The additional instructions for the disk are simply accessed from the Program Pak. (We were successful here.)When the machine is turned on, the necessary links to the "operating system" require only about 2K of user RAM. (This means the user has 11,047 bytes of RAM memory in a 16K system to enter and run his/her program after executing a PCLEAR 1.)



ROM Cartridge plugged into Color Computer

Another nice thing about the Color Disk System, besides the Input/Output functions to disk and the necessary string conversions which are required to facilitate those functions, on power-up, no diskette need be in any drive. (Since the operating system is in ROM, there is no need to have a TRSDOS diskette in drive #0 from which to load the operating system!) In addition, when accessing drive #1 (the standard cable supplied will handle up to 2 drives—an optional cable will handle up to 4 drives, but that's another story), you don't need a diskette in drive #0.

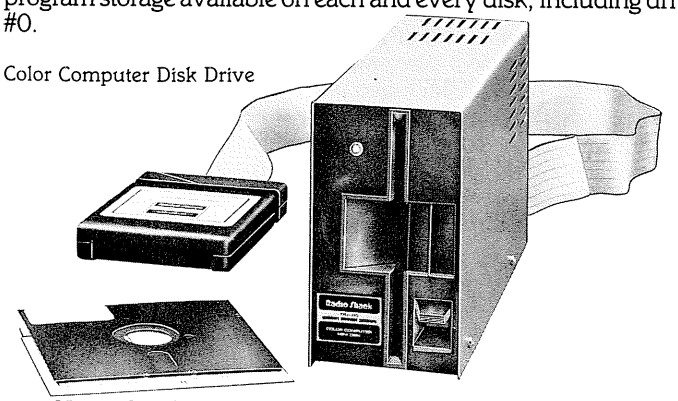
We've added a few commands and changed some others. For instance, to format a diskette, instead of using the traditional "FORMAT" instruction, we have chosen to use "DSKINI" which stands for DiSK INItialization. Simply add a drive number at the end and the diskette in that drive will be formatted. Again, if you want to use drive #1, you do not need a diskette of any kind in drive #0.

The "FREE" command has been changed to "PRINT FREE (b)" which will tell you the number of grans (amount of disk space) free on that diskette. You might need some explanation here, but read on and it will be explained.

We've even added a couple of instructions to enable you to read tracks/sectors or write track/sector directly without using the directory. (No room for an explanation of these instructions, maybe a later article.)

As was desired, there are no operating instructions or utilities on the diskette, since they are all in the Program Pak. The disk drive used is a 35 track, double density, 5 1/4" floppy drive. (Similar to those used by the Model III, but not the same units.) Each diskette, when it is initialized, will offer the user 34 tracks for storage with one track being used for the directory. (So the drive knows where on the diskette the information is stored.) Each of the 34 tracks is divided into 18 sectors. Each sector is divided into 256 bytes. This means that the user has 156,672 bytes of data/program storage available on each and every disk, including drive #0.

Color Computer Disk Drive

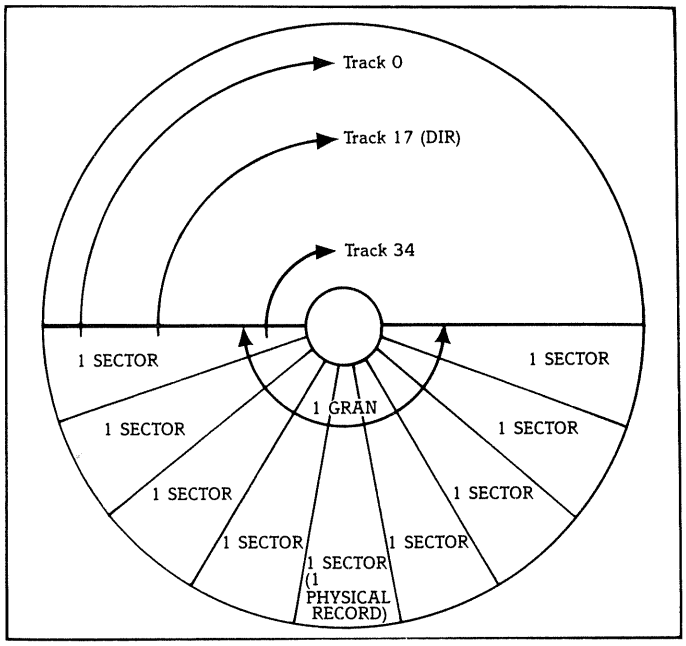


Now, for the explanation of "PRINT FREE (b)." Each track contains 2 grans (9 sectors of 256 bytes each or a total of 2304 bytes per gran). When you save a program or data file, the system allocates 2304 bytes (1 gran). When that space is used up by the same program or data file, it gives another 2304 bytes. For each program, it will allocate 1 gran (or more if necessary), even if the program is only 10 bytes long. When you enter a PRINT FREE command, it tells you the number of unallocated grans left on the diskette for the drive you specify with the (b). An initialized diskette with no data or programs will have 68 grans left (34 tracks at 2 grans per track).

(Continued on Page 28)

Color (From Page 27)

One of the 35 tracks contains the directory, GAT and HIT tables (tells how many grans allocated and their location), and some extra space for future use.



Diskette Equivalent Space Table

Bytes	Sectors	Granules (Gran)	Tracks	Disk
256	1	—	—	—
2,304	9	1	—	—
4,608	18	2	1	—
162,075	630	70	35	1

A PHYSICAL RECORD IS DEFINED AS 1 SECTOR

Look for deliveries to start around the first part of November in the stores. Order now so you will be sure to get one as soon as they are available.

For some of you, this information might seem a little mind boggling. Take heart, the system was designed with you in mind. You should find using the disk extension to the Extended BASIC Color Computer a most pleasant and rewarding experience. We have written a manual (included with your first drive) which will take you through the operations of the disk system and have you manipulating programs and examining files in no time. There are even sections for people who want to learn only the bare necessities for running applications software or who are already familiar with our other disk machines and only want to know the command syntax of the instructions.

Now how's that for service with a smile. . . Until next month.

Auto Repeat

Alexander Benenson New York, NY

Enclosed is a diagram of the Color Computer's keyboard matrix, showing the keys that are scanned by each address, and how they affect the contents of that address.

ADDR.	KEYS						
338	@	H	P	X	0	8	CLEAR
339	A	I	Q	Y	1	9	ENTER
340	B	J	R	Z	2	:	
341	C	K	S	↑	3	;	
342	D	L	T	↓	4	,	
343	E	M	U	←	5	—	
344	F	N	V	→	6	.	
345	G	O	W	space	7	/	
	-1	-2	-4	-8	-16	-32	-64 Change of contents
	254	253	251	247	239	223	191 Contents if pressed

No Key Pressed = 255

This information lets us use PEEK as a way to scan the keys, like INKEY$. But, PEEK has an advantage over INKEY$; PEEK can read a key over and over, without you having to release the key and press it again. Also, PEEK can detect more than one key being pressed at a time. Let's see how this works. Suppose you want to see if A and Q are being pressed. The chart says that both A and Q are scanned by address 339. At the bottom of A's column, it says that the memory contents are reduced by 1 if any key from that column is pressed. For Q the reduction is 4. 255-1-4 = 250. So, 250 is the result we should get in 339 if both A and Q are pressed. Try it! Type in:

```
10 PRINT PEEK(339): GOTO 10
```

and RUN it. You will see that if you don't touch any keys, 255 is printed. If you hold down A, 254 will be printed until you release A key, then 255 will again be printed. Try pressing both the A and Q keys to see if you really get 250. Try other keys, and change the 339 to one of the other addresses on the chart to see what happens. Try pressing the 'fire' buttons if you have joysticks.

Since 255 means that no key is being pressed, maybe you can fool the computer into thinking that we have released a key when we haven't, so that INKEY$ will keep returning values, even if hold the key down. Type in this program:

```
1Ø A$=INKEY$
2Ø PRINT A$;
3Ø FOR N=338 TO 345
4Ø POKE N, 255
5Ø NEXT
6Ø GOTO 1Ø
```

Line 10 scans the keyboard and returns the value into A$. Line 20 prints the value found. Lines 30-50 'fool' the computer by POKEing 255 (meaning 'no key pressed here') into the 8 keyboard addresses. Line 60 starts the loop over.

RUN the program. Notice how any key you press is printed until you let go of it. Try the arrow keys, ENTER, etc. Use shift, and put the computer into uppercase/lowercase mode with SHIFT 0 to get lowercase (reverse) letters. Why do I mention SHIFT? Look at the chart again, and see how SHIFT is not there, and how all symbols you get by pressing SHIFT are not there either? The computer gets shifted entries by saying to itself: Okay, the 1 key has been pressed. Is SHIFT pressed too? If it is, I will print a !. If it is not I will just print a 1. To prove this to yourself, type in the first program again and see how pressing SHIFT by itself or with other keys does not affect anything, no matter which address you PEEK at.

With this information, I am sure anyone can get some great ideas for many program applications. It is especially useful in action games for Color Computers without joysticks, and even you 'stuck up' people out there with joysticks can see that 4 variable inputs and 2 on/off inputs are not always enough.

Here is a simple target game. Press the space bar to fire, use the arrow keys to move your base.

```
Ø1Ø 'SHOOT'EM UP
Ø2Ø 'BY Alexander Benenson
Ø3Ø 'Create UFO's laser base
Ø35 'Delete REM statements to increase speed
Ø4Ø S$=CHR$(129)+CHR$(143)+CHR$(143)+CHR$(143)
    CHR$(13Ø)
Ø5Ø E$=CHR$(128)+CHR$(128)+CHR$(128)+CHR$(128)
    CHR$(128)
Ø6Ø R$=CHR$(177)+CHR$(191)+CHR$(191)+CHR$(191)
    CHR$(178)
Ø7Ø O$=CHR$(241)+CHR$(255)+CHR$(255)+CHR$(255)
    CHR$(242)
Ø8Ø B$=CHR$(241)+CHR$(251)
    : BE$=CHR$(128)+CHR$(128)
```

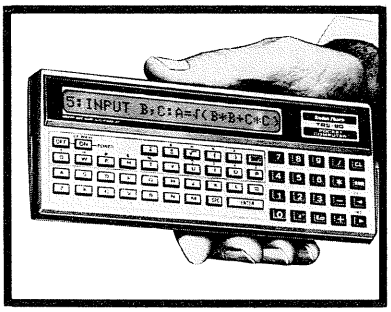
(Continued on Page 29)

Auto Repeat (From page 28)

```
ØØ BL$=CHR$(161)+CHR$(175)+CHR$(175)+CHR$(175
    CHR$(162)
1ØØ A=247
    : CLSØ
    : T=495
    : PRINT@T, B$;
    : I1=1
    : I2=-1
    : FS=2
    : P1=Ø
    : P2=91
11Ø OT=T
12Ø 'Check for left arrow
13Ø IF PEEK(343)=A THEN T=T-1
14Ø 'Check for right arrow
15Ø IF PEEK(344)=A THEN T=T+1
16Ø 'Move laser base
17Ø IF T>5Ø9 THEN T=48Ø
18Ø IF T<48Ø THEN T=5Ø9
19Ø IF OT<>T THEN PRINT@OT,BE$;
    : PRINT@T, B$;
2ØØ 'Move UFOs
21Ø O1=P1
    : P1=P1+I1
    : IF P1>26THEN P1=Ø
    : I1=RND(3)
22Ø O2=P2
    : P2=P2+I2
    : IF P2<64 THEN P2=91
    : I2=-RND(3)
23Ø PRINT@O1, E$;
    : PRINT@P1, S$;
24Ø PRINT@O2, E$;
    : PRINT@P2, BL$;
25Ø IF F=1 THEN 3ØØ
26Ø 'Check for space-bar
27Ø IF PEEK(345)=247 THEN F=1
    : FY=3Ø
    : SOUND 1, 1
    : FX=(T-479)*2
28Ø IF F=Ø THEN 11Ø
29Ø 'Move laser
3ØØ OF=FY
    : FY=FY-FS
    : RESET(FX,OF)
31Ø IF FY<Ø THEN F=Ø
    : PRINT@T, B$;
    : GOTO 11Ø
32Ø 'Hit it?
33Ø IF POINT(FX,FY)=1 THEN P=P1
    : Z$=S$
    : GOTO 38Ø
34Ø IF POINT(FX,FY)=3 THEN P=P2
    : Z$=BL$
    : GOTO 38Ø
35Ø SET(FX,FY,7)
36Ø GOTO 11Ø
37Ø 'UFO explosion
38Ø FOR N=1 TO 3
    : PRINT@P, Z$;
39Ø FOR Q=1 TO 5Ø
    : NEXT
    : PRINT@P, R$;
    : SOUND 1ØØ, 1
4ØØ FOR Q=1 TO 5Ø
    : NEXT
    : PRINT@P, O$;
    : SOUND 15Ø, 1
41Ø FOR Q=1 TO 5Ø
    : NEXT
    : SOUND 2ØØ, 1
    : NEXT
42Ø PRINT@T, B$;
    : F=Ø
    : GOTO 11Ø
```

Conjugation (From Page 19)

```
25Ø GOSUB 34Ø
    : PRINT E$(1); T2$
26Ø POKE 15616 + LEN(E$(1)) + LEN(T2$), 28
27Ø FOR I=2 TO 8
    : PRINT D$(I); E$(I); T2$
28Ø POKE 15552 + (I*64) + LEN(D$(I)) + LEN(E$(I)) +
    LEN(T2$), 28
29Ø NEXT
    : GOTO 44Ø
3ØØ GOSUB 34Ø
    : PRINT E$(1); T2$; "I"
31Ø FOR I=2 TO 8
    : PRINT D$(I); E$(I); T2$; "I"
    : NEXT
    : GOTO 44Ø
32Ø GOSUB 34Ø
    : PRINT E$(1); T2$; "U"
33Ø FOR I=2 TO 8
    : PRINT D$(I); E$(I); T2$; "U"
    : NEXT
    : GOTO 44Ø
34Ø CLS
    : PRINT TAB(14) "THE PAST CONJUGATED FORMS OF " V$
35Ø PRINT
    : PRINT "PAST"
    : PRINT
    : RETURN
36Ø IF T$="RE" THEN 39Ø
37Ø GOSUB 42Ø
    : PRINT A$; V$; F$(1)
38Ø FOR I=2 TO 8
    : PRINT D$(I); V$; F$(I)
    : NEXT
    : GOTO 44Ø
39Ø GOSUB 42Ø
    : PRINT A$; LEFT$(V$,LEN(V$)-1); F$(1)
4ØØ FOR I=2 TO 8
    : PRINT D$(I); LEFT$(V$, LEN(V$)-1); F$(I)
41Ø NEXT
    : GOTO 44Ø
42Ø CLS
    : PRINT TAB(13) "THE FUTURE CONJUGATED FORMS OF "
    V$
43Ø PRINT
    : PRINT "FUTURE"
    : PRINT
    : RETURN
44Ø PRINT@798, "PRESS"
45Ø PRINT@899, "(1) TO SEE NEXT FORM, (2) TO START
    OVER, (3) TO END PROGRAM"
46Ø Q$=INKEY$
    : IF Q$="2" THEN RUN
47Ø IF Q$="3" THEN CLS
    : END
48Ø IF Q$="1" THEN 49Ø ELSE 46Ø
49Ø C=C+1
    : IF C>3 THEN C=1
5ØØ ON C GOTO 24Ø,36Ø,15Ø
51Ø DATA "E", "ES", "E", "E", "ONS", "EZ", "ENT",
    "ENT"
52Ø DATA "IS", "IS", "IT", "IT", "ISSONS", "ISSEZ",
    "ISSENT", "ISSENT"
53Ø DATA "S", "S", "", "", "ONS", "EZ", "ENT", "ENT"
54Ø DATA "JE ", "TU ", "IL ", "ELLE ", "NOUS ", "VOUS
    ", "ILS ", "ELLES "
55Ø DATA "J'AI ", "AS ", "A ", "A ", "AVONS ", "AVEZ
    ", "ONT ", "ONT "
56Ø DATA "AI", "AS", "A", "A", "ONS", "EZ", "ONT",
    "ONT"
```

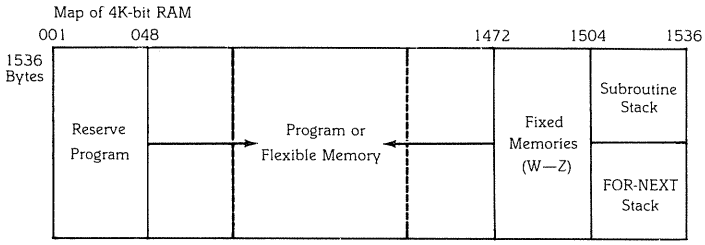


Pocket Computer

Product Line Manager's News

RAM Memory in the Pocket Computer

Random Access Memory in the Pocket Computer consists of three 4K bit CMOS chips plus 1K bits of memory in each of the three display chips. Lets look first at the memory in the display chips:



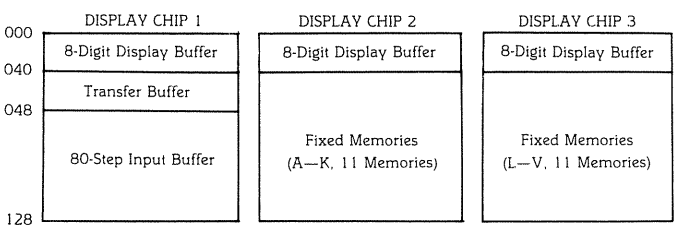
The 1K bits in each display chip are divided into 128 eight-bit bytes. The first 40 bytes of memory in each of the display chips is used for two purposes: First, as an 8-character display buffer, giving us memory for the 24 characters (8 characters × 3 chips) which can be displayed at any one time. Each byte represents one of the five vertical segments in a displayed character. The second use is as a buffer or working memory for results during arithmetic operations (which should explain why the display goes blank during processing).

In display chip 1, we first have the 40 byte display buffer. The next 8 bytes are used as the transfer buffer for information between CPU I and CPU II. The remaining 80 bytes are used as the 80-step input buffer. This input buffer is used to hold:

1. Any information which is entered through the keyboard (Up to the maximum 80 steps).
2. When information is being displayed, the full 80 steps are stored here by CPU I. CPU II then selects the information to be displayed. This allows us to scroll through a program line.
3. When an arithmetic operation is to be performed, CPU I uses this buffer to hold the procedure. CPU II can then perform the operation according to the procedure stored by CPU I.
4. When information is being recorded or read from tape, this buffer is used for information transfer. We might assume, that this same buffer is used during output to the Pocket Printer.

In display chip 2, we again have the 40 byte display buffer. The remaining 88 bytes are used as fixed memory locations A—K. Remember that each fixed memory location uses 8 bytes (8 bytes × 11 memories = 88 bytes) of memory.

In display chip 3, we again have the 40 byte display buffer. The remaining 88 bytes are used for fixed memories L—V.



Let's look at the three 4K bit RAM chips. Each of these chips is configured as 512 8-bit bytes, giving us a total of 1536 bytes. The first 48 bytes are used for reserve program memory. Bytes 1473—1504 (32 bytes) are used for fixed memories W—Z. Bytes 1505—1536 are used for the Subroutine stack, and the FOR-NEXT stack.

The remaining memory is used for a combination of program storage and flexible memories. Programs are stored beginning at memory location 49 and build toward location 1472. Flexible memories are stored beginning at location 1472 and build toward location 49.

Some Details:

Characters are displayed on the Pocket Computer using a 5 × 7 dot matrix. Since we are only displaying 7 dots vertically, bit 8 of each byte is ignored. In the remaining 7 bits, a 1 represents a dot to be turned on, and a 0 represents a dot to be left off.

Example: Here is how the numeral "4" looks on the PC display

```
H1 00010
H2 00110  H1—H8 represent the 8 bits in a single
H3 01010        byte of display buffer memory.
H4 10010
H5 11111  S1—S40 are the 40 bytes in a single
H6 00010        display chip.
H7 00010
H8 XXXXX  H8—not used for display

   SSSSS . . . S . . . S
   12345       1 . . . 4
                 0     0
```

Each of the vertical columns is one byte in memory:

ADDRESS								DISPLAY SEGMENT								
A8	A7	A6	A5	A4	A3	A2	A1	H8	H7	H6	H5	H4	H3	H2	H1	
0	0	0	*	0	0	0	0	X	0	0	1	1	0	0	0	S1
0	0	0	*	0	0	0	1	X	0	0	1	0	1	0	0	S2
0	0	0	*	0	0	1	0	X	0	0	1	0	0	1	0	S3
0	0	0	*	0	0	1	1	X	1	1	1	1	1	1	1	S4
0	0	0	*	0	1	0	0	X	0	0	1	0	0	0	0	S5

After the appropriate display chip is selected, the display information is addressed as follows:

The display buffer is divided into three blocks with 16 bytes in each of the first two blocks and 8 bytes in the third block. Address line A8—A6 are used to select the block. 000 selects the first block, 001 the second and 010 the third block.

Within each block, address lines A4—A1 select one of the 16 possible segments (S1-S16, S17-S32, S33-S40). This leaves address line A5 unaccounted for.

As you may be aware, the Pocket Computer CPUs are 4 bit CPUs. Because of this there are only four data lines in the computer. In order to be able to move 8 bit information, the information must be moved 4 bits at a time. In the display buffers, bit 5 is used to select between the upper 4 bits of data (A5 = 1, H8-H5) and the lower four bits (A5 = 0, H4-H1). This means that to move the information required to display a single character (5 bytes) requires 10 data movements. To get the information into S1 for the first column of our "4," address 00000000 is used to move 1000 into H4-H1, and address 00010000 is used to move 0001 into H8-H5.

Satellite

Doug Swank Waltham, Massachusetts

My compliments on the Newsletter—it gets better with each issue. I have both a Pocket Computer and a Color Extended machine, and get a lot of useful info from each of the columns.

Would like to pass along a short program I did to calculate antenna aiming for satellite and ground stations. It is just a textbook exercise in spherical trig, no copyrights possible, so feel free to use it if you like.

This is written for the PC, but changing the I/O formats should make it usable on any machine with BASIC and the trig functions. (Note that 'arcsin' and 'arccos' must be derived from other functions in the Color Computer.)

Lines 100-160 produce true bearing and elevation to any geosynchronous satellite. (They are always over the equator, so we only need their longitude.) Program starts with **[SHFT S]**; **[SHFT X]** will rerun for another satellite.

Since the program works between two spherical systems and the math gets messy, a few simplifications have been made. Result is that the program only works for North America (Above the Equator.), and accuracy is about + or - 0.2 degree. It is close enough to get a signal on the antenna for fine tuning. Here on the East Coast many of the popular satellites are low in the sky. A simple plumb-line and protractor can be useful to check that the proposed site is not blocked by trees or buildings. Note that bearing (Azimuth) angles are true, not compass bearings.

Lines 200-325 produce true bearing and distance for any two points on earth. North Latitudes and West Longitudes are inserted as positive numbers, East and South as negative. Bearings are exact, distances have a small error since the earth is not a perfect sphere. (Typical error less then 1 %.) Starts with **[SHFT A]**, rerun for new destination is **[SHFT Z]**. Inputs are in Degree/Min/Sec format.

This one is simply a solution of the Law of Cosines to produce a Great Circle Course, and might be useful to pilots and sailors as well as radiocommunications users. A save to tape will record location data as well as the program, if it has been loaded.

Thanks again for the good work on the Newsletter.

```
1 REM "D.L.SWANK 7/81"
100 REM"SATELLITE ANT.AIMING
110 "S"
    : X=0.15091
115 INPUT "E/S LAT.="; U
120 INPUT "E/S LONG.="; T
125 "X"
    : INPUT"SAT.LONG.="; S
130 V=S-T
135 W=   ((1+(X*X))-(2*X*COS V*COS U))
140 Y=ASN (((COS U*COS V)-X)/W)
145 Z=270-ACS(SIN V/(W*COS Y))
149 PRINT USING "####."; "SAT AT"; S; "DEG: "
150 PRINT USING"####.#"; "AZ="; Z
155 PRINT "EL="; Y
160 END
200 REM "*TERRESTRIAL ANT.AIMING"
250 "A"
    : INPUT "STA.LAT="; M
255 M=90-DEG M
260 INPUT "STA.LONG.="; N
265 N=DEG N
270 "Z"
    : INPUT "DEST.LAT.="; O
275 O=90-DEG 0
280 INPUT "DEST.LONG.="; P
285 P=DEG P
    : L=N-P
290 Y=ACS ((COS M*COS O)+(SIN M*SIN O*COS L))
295 Z=ACS ((COS O-COS Y*COS M)/(SIN Y*SIN M))
300 IF SIN L<0 THEN Z=360-Z
305 PRINT USING "####.#"; "AZ="; Z; " TRUE"
310 Y=60*Y
315 PRINT USING "######"; "DIST.="; Y; " NMI"
325 END
```

Easter

John T. Livingston P.E.
Avon-By-The-Sea, New Jersey

Having received one of your TRS-80 Pocket Computers for Christmas I have been having a grand time programming anything that would stand still!

At sometime in the past I ran across an empirical method for determining the date Easter falls on for any given year, a date that has always been determined by referring to religious tables. This looked like an interesting program subject so the following is the result. Incidentally I checked this against tables from 1900 to 2089 and, surprise!, it agreed!

```
10: I."   YEAR WANTED - "; Y
20: N=Y-1900
30: Z=N/19
40: V=INT Z
50: A =(Z-V)*19
     : A=A+.05
     : A=INT(A*10)/10
60: B=((7*A)+1)/19
     : B=INT B
70: T=((11*A)+4-B)/29
80: M=(T-(INT T))*29
90: Q=N/4: Q=INT Q
100: D=(N+Q+31-M)/7
110: W=(D-(INT D))*7
     : W=W+.05
     : W=INT(W*10)/10
120: E=25-M-W
     : E=E+.05
     : E=INT(E*10)/10
130: IF E<=0 G. 150
140: P."   EASTER IS APRIL   "; E
     : G.  10
150: G= 31 +E
160: P."   EASTER IS MARCH   "; G
     : G.10
```

I hope to see more interesting programs in your publication and hopefully other manuals devoted to the PC. Have fun!!

Ft. Worth Scene (From Page 32)

I mentioned earlier that our mail goes through several departments before we get it. The first department to see the mail for the newsletter is our data processing department. They scan the mail for subscription requests or problems and handle those letters at their level. Any letters with material that does not pertain directly to subscription problems is forwarded to Computer Customer Services (CCS). The CCS people handle most of the questions and all of the problems that you write about. The people in CCS have those answers and can reply much faster than we could here at the Newsletter. Finally, any letters which contain material for the newsletter are forwarded to us. We have found that this procedure gets each class of letter the fastest and best possible response without burying anyone. The system is not perfect, and some of the letters seem to fall into black holes never to be seen again, but we are trying to eliminate the black holes as we find them.

Bruce W. Elliott - Editor
Linda Miller - Writer
Richard Conner - Production Art

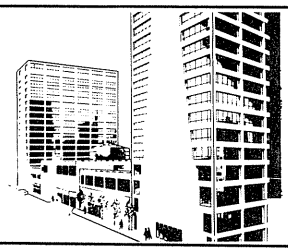
Radio Shack®
MICROCOMPUTER NEWS
P.O. BOX 2910
FORT WORTH, TEXAS 76113-2910

BULK RATE
U.S. POSTAGE
PAID
Radio Shack
A Div. of Tandy Corp.

ADDRESS CHANGE
☐ Remove from List
☐ Change as shown
Please detach address label and mail to address shown above.

IF UNDELIVERABLE DO NOT RETURN

Fort Worth Scene



This month we are happy to add a column from Dow Jones to our list of features. We are looking forward to a lot of timely information from the Dow Jones people. The decision to add the Dow Jones column was made a little late in the cycle of putting this month's newsletter together, and the Dow Jones people have been great about meeting our deadline on short notice.

We do want to mention, that while we intend for the various columns to appear monthly, there may be some months in which one or more of the articles will not appear, or in which the material may be provided by an alternate source. One example is the VisiCalc article. Currently the people at Personal Software are providing us with an article every other month. This month we were able to fill in with the excellent article on sorting VisiCalc data by Mr. Spencer. For November we expect to have another article from the VisiCalc people, and for December, we are not sure yet.

We hope you noticed! After five months of 24 page newsletters, this one is 32 pages. When we began considering a size increase from 24 pages, one of the questions was "Where is the material going to come from?". The answer is that much of the information will (and does) come from you, our readers. We will do our part, by continuing to prod the Product Line Managers for ideas and articles, and by getting the outside articles together and edited each month.

You as readers also need to do your part (and in recent months the response to the newsletter has been tremendous). If you have a favorite program, a helpful hint, or a new way to do something, send it to us. We will NOT use it the next month, but we will try to use it in a future issue. We have recently instituted a policy of sending a letter to each individual who submits information for publication. We will try not to miss anyone. Unfortunately, our mail goes through several departments before it finally reaches us (yes, there is a good reason) and occasionally your letter or program may not reach us. If you are submitting material for publication and you do not get some sort of comunication from us in 8 to 12 weeks after you mailed the information, drop us another line (maybe even a copy of the original material).

What we are looking for is material which will benefit other TRS-80 users. Some of our users are very new to computers. For these individuals we will print what may appear to be "trivial" information or programs. Others of you are quite sophisticated and you may have a program or technique which will appeal to a small segment of our readership. We will try to publish information for both types of reader, as well as for all the readers in between these extremes. If you think the information and programs we are presenting don't fit what you need, please do two things - first, tell us; second, send us material of the type that does fit your needs. If each reader will provide us with a single article, then each newsletter will have an excellent chance of containing material which fits the needs of a wide range of our readers.

Over the next few months we would like to introduce several new types of articles (hopefully on a regular basis). Specifically we will be looking for information on or about: Advanced Programming Techniques (Merging VisiCalc, Profile II and Scripsit to produce reports, etc.), Application Stories (How are programs and computers being used by real people), BASIC command explanations and summaries, Graphics Techniques (both video and hardware), Languages (how are you using Tiny Pascal, or Compiler BASIC?), Machine Languages (both Z-80 and 6809), Tutorials, and more. If you have material you would like to share with other TRS-80 users, send it. If you have specific suggestions for articles, send them too.

(Continued on Page 31)

TRS-80 Microcomputer News is published monthly by Radio Shack, a division of Tandy Corporation. A single one year subscription is available free to purchasers of new TRS-80 Microcomputer systems with addresses in the United States and APO or FPO addresses. Subscriptions to other addresses are not available. The subscription rate for renewals and other interested persons is twelve dollars ($12.00) per year, check or money order in U.S. funds. All correspondence related to subscriptions should be sent to: Microcomputer News, P.O. Box 2910, Fort Worth, Texas 76113-2910.

Back issues of Microcomputer News are not available.

The TRS-80 Newsletter welcomes the receipt of computer programs, or other material which you would like to make available to users of TRS-80 Microcomputer systems. In order for us to reprint your submission, you must specifically request that your material be considered for reprinting in the newsletter and provide no notice that you retain copyrights or other exclusive rights in the material. This assures that our readers may be permitted to recopy and use your material without creating any legal hassles.