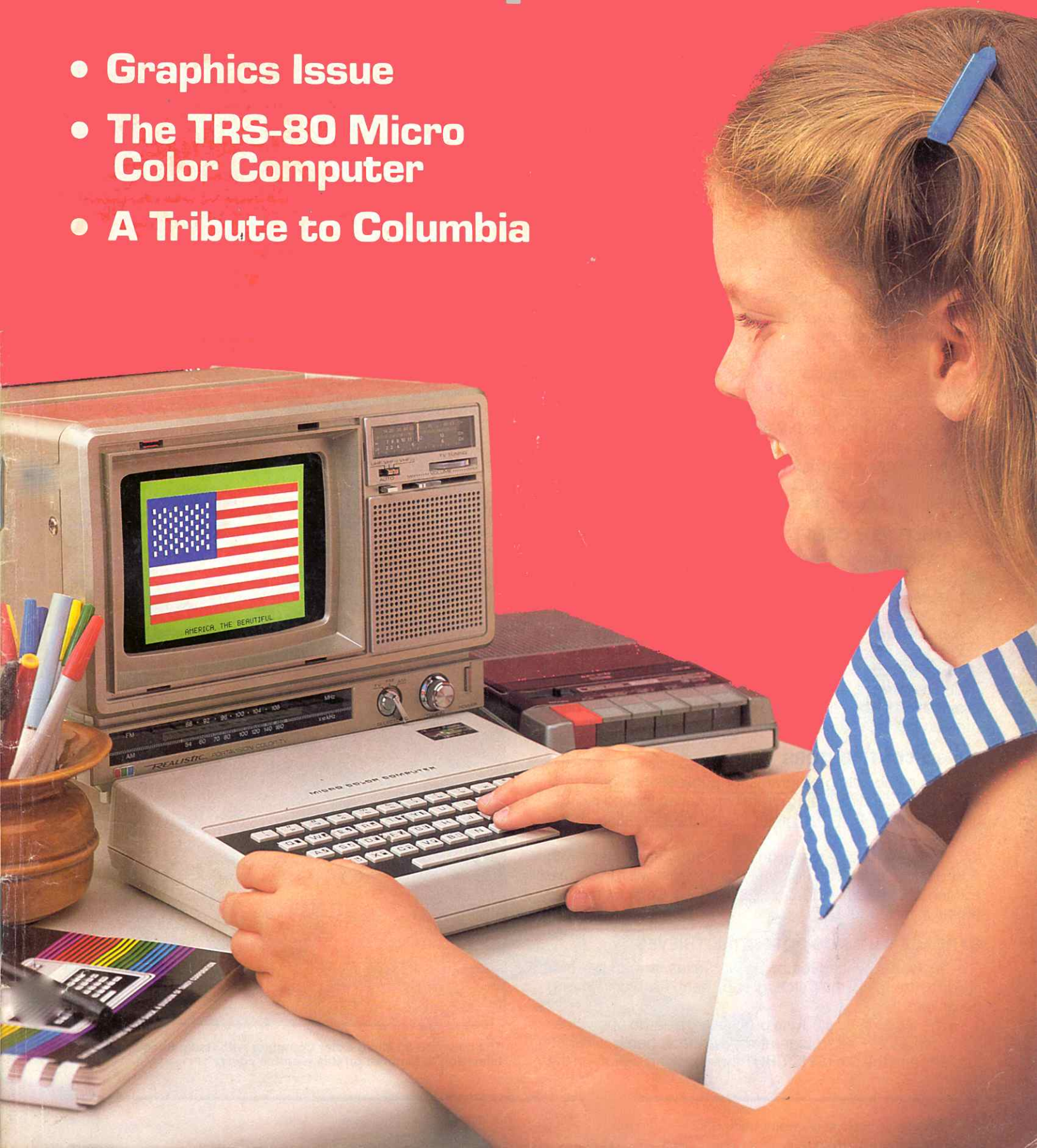
Volume 5, Issue 6 June, 1983
$1.50

TRS-80® Microcomputer News

- Graphics Issue
- The TRS-80 Micro Color Computer
- A Tribute to Columbia



Fort Worth Scene

First we want to thank all those who submitted graphics programs. We received many more than available space permits us to publish. When we started looking through the reader submitted programs, we found—as usual—a disproportionate number of programs for the Color Computer—especially in Extended Color BASIC. For Model I/III owners, "Tribute to Columbia" provides very nice graphics and a tutorial on the Columbia space program. We hoped to receive at least one or two programs written for the Model II or III graphics boards, but after searching through the files, we came up empty handed. We're still waiting for those programs to arrive.

Although not a graphics program, a Model II/12 program for listing a BASIC program in an attractive format creates great looking print outs.

ANOTHER NEW COMPUTER

1983 has been a great year for TRS-80s. We've seen the introduction of the Models 4, 12, 100, and the Pocket Computer 4. Still another new computer, the TRS-80 Micro Color Computer (Model MC-10), is introduced in this issue. Color, graphics, and BASIC programming capability along with a very modest price make the MC-10 a great buy for anyone wanting to own a real computer without making a major investment. The MC-10 screen was created by a program which was written for the Color BASIC Computer, but it runs on the MC-10 without any modifications.

TO ALL CONTRIBUTORS

When submitting programs for publication, please indicate which machine the program was written for and include any documentation that might clarify the operation of the program.

Often we receive media on a 5 1/4" diskette or a tape with no indication of which computer the program will run on. We scratch our heads wondering if it is for the Color Computer, Model I, or Model III.

GARBAGE OUT AIDS POTENTIAL THIEVES

Mr. C. Laurence Gott of Dallas, Texas, a member of a Neighborhood CrimeWatch group, sent us the following news item.

> The Dallas Police advise purchasers of large appliances such as stoves, refrigerators, washers, home computers, and the like to NOT PUT their packing carton on the curb (or wherever) with the trash to be picked up in the normal course of garbage collection. This gives observant looters and others working for black-marketeers information regarding new major items in your house. Rather, take the packing case immediately to the dump or to a large neighborhood dumpster (with permission from the owner, of course.)

The January cover of *TRS-80 Microcomputer News* inspired Mr. Gott to send us the above information, and we publish it in the hope that by making our readers more aware of potential threats, you will not become a victim.

MAGAZINES

Below are five magazines of special interest to TRS-80 owners that we believe have editorial content of high quality and will be of use to our customers.

80-US (Covers all TRS-80's)
3838 South Warner Street
Tacoma, WA 98409
(206)475-2219

Color Computer Magazine
Highland Hill
Camden, ME 04843
(207)236-9621

Color Computer Weekly
P.O. Box 1355
Boston, MA 02205

Rainbow (Covers the TRS-80 Color Computer)
5803 Timber Ridge Dr.
Prospect KY 40059
(502)228-4492

two/sixteen magazine
P.O. Box 1216
Lancaster, PA 17603
(717)397-3364

The new TRS-80 Micro Color Computer with Radio Shack's 5" color television is featured on this month's cover.

TRS-80 Microcomputer News
Volume 5 Issue 6
JUNE 1983

TRS-80 Microcomputer News is published monthly by Radio Shack, a division of Tandy Corporation, One Tandy Center, Fort Worth, Texas U.S.A. 76102.

TRS-80 Microcomputer News is published monthly by Radio Shack, a division of Tandy Corporation. A single six month subscription is available free to purchasers of new full size TRS-80 Microcomputer systems with addresses in the United States, Puerto Rico, Canada and APO or FPO addresses. Certain smaller TRS-80 Microcomputers will not include this free subscription. Subscriptions to other addresses are not available.

The subscription rate for renewals and other interested persons with U.S., APO or FPO addresses is twelve dollars ($12.00) per year, check or money order. Single copies of the Microcomputer News may be purchased from Radio Shack Computer Centers or Computer Departments for $1.50 suggested retail each.

The subscription rate for renewals and other interested persons with Canadian addresses is Fifteen dollars ($15.00) 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.

Retail Prices in this newsletter may vary at individual stores and dealers. The company cannot be liable for pictorial and typographical inaccuracies.

Back issues of Microcomputer News prior to January, 1981 are available through your local Radio Shack store as stock number 26-2115 (Suggested Retail Price $4.95 for the set). Back issues of 1981 copies are available as stock number 26-2240 (Suggested Retail Price $9.95 for the set).

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.

Material for publication should be submitted on magnetic media (tape, disk, or CompuServe). If you submit material on tape or disk, and it is accepted for publication, we will send you two cassettes or diskettes for each one you sent us. Cassettes will come from our box of mixed blank cassettes. If you submit material on CompuServe, and we think we may use the material, we will extend your Microcomputer News subscription by six months for each article accepted. If you are submitting material over CompuServe, please include your name and address or your subscription number so we can find you. If the material is very short, send it to us in E-Mail. If you have more than a few lines, you need to place the material in the ACCESS area of CompuServe and then let us know it is there by leaving a message on E-Mail.

Material may be submitted by mail to P.O. Box 2910, Fort Worth, Texas 76113-2910, or through CompuServe. The Microcomputer News' CompuServe user ID number is 70007,535.

Programs published in the Microcomputer News are provided as is, for your information. While we make reasonable efforts to ensure that the programs we publish here work as specified, Radio Shack can not assume any liability for the accuracy either of the programs themselves or of the results provided by the programs.

Further, while Microcomputer News is a product of Radio Shack, the programs and much of the information published here are not Radio Shack products, and as such can not be supported by our Computer Customer Service group. If you have questions about a program in the Microcomputer News, your first option is to write directly to the author of the program. When possible, we are now including author's addresses to facilitate communications. If the address is not published, or if you are not happy with the response you get, please write us here at Microcomputer News. We will try (given the limited size of our staff) to find an answer to your question and, in many cases, will publish the answer in an up-coming issue of Microcomputer News.

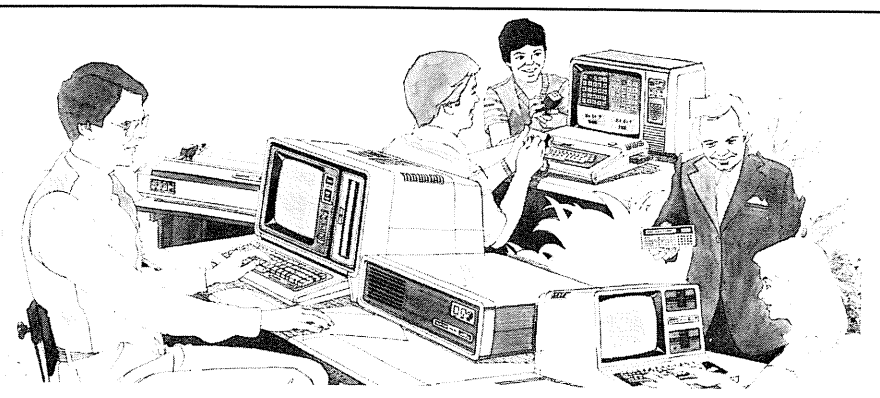
Trademark Credits

CompuServe™	CompuServe, Inc.
CP/M®	Digital Research
Dow Jones™ NEWS/RETRIEVAL Service®	Dow Jones & Co., Inc.
LDOS™	Logical Systems, Inc.
VisiCalc®	VisiCorp, Inc.
XENIX™	Microsoft
Program Pak™	Tandy Corporation
SCRIPSIT™	Tandy Corporation
TRSDOS™	Tandy Corporation
TRS-80®	Tandy Corporation

TRS-80® Microcomputer News

Contents:

Prices shown in TRS-80 MICROCOMPUTER NEWS are in U.S. Funds.



Scripsit Plotter Driver

The Scripsit Plotter Driver package (Catalog no. 26-4536, suggested retail $49.00) is a fast, easy way to use the Radio Shack Multi-Pen Plotter (Catalog no. 26-1191, suggested retail $1995.00) to produce colorful, striking graphics from your Model II, 12, or 16 in a relatively short period of time. Even if you have never used a graphics package in the past, you can create professional looking graphics on your first attempt. The six-pen plotter can be used to make reports with eye-catching charts and diagrams, journal articles and professional papers with graphs and block diagrams built right into the text, overhead transparencies for lectures and sales presentations, and any number of posters and fliers.

Let's take a look at the features of the Scripsit Plotter Driver. There are several features that control how your final document will look. The first of these is Format. Format tells the plotter that the document will be a vertical or a horizontal document, in much the same way that the Create New Document menu in Scripsit 2.1.0 does. You Format a document by the use of Scripsit Printer Control Codes. Merely type CTRL X (the two keys together, in either capital or lower case), then the letter "F" for format, followed by either a "V" or an "H" for Vertical or Horizontal.

The second feature that can affect how your document will look is Size. Size determines how large your characters will be. This can vary greatly depending on placement on the page, format used, and type of usage. For example, you may want to have one size for text within a diagram, another size for captions, and yet another size for titles and subheadings. Keep in mind that the plotter driver can produce an entire document consisting of text as well as graphics. When you want to define the text size, type a CTRL X, then the letter S for Size, followed by a number between 1 and 9. Here is an example of size 1 through 9 text.

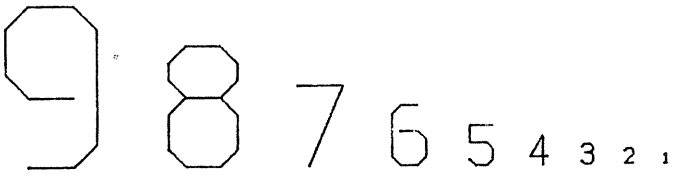


Figure 1.

For convenience, the Scripsit Print Control Codes do not take up any physical space in the document. When you want to tell the plotter which of the six pen colors to use, type a CTRL X then a letter P for Pen, followed by a number between 1 and 6. The default pen color is 1 (black).

Now that you have defined the basics of your document, you will want to decide on the types of graphics you want to produce. You will probably rely on the Tandygraph program to produce any line graphs, pie charts, or bar graphs for your business reports and professional papers, but such items as an organizational chart or a flowchart for time-series analysis or diagramming a computer program is beyond the Tandygraph's ability, and you will have to rely on the features of the Scripsit Plotter Driver. The plotter driver has two commands that make it easy for you to get started. The first is Box. Box lets you define a box of any size and shape wherever you desire on the page. To define the upper left corner of a box type CTRL X, then the letter B, followed by a number between 1 and 9. Scripsit will only allow you to define up to 9 boxes at one time. You can now fill that box with text, other boxes, or other graphics. To define the lower right corner of the box, type CTRL X, then the letter B again, followed by the same box number you used to start the box.

The next feature you will want to be familiar with is Line. Drawing a line is very similar to creating a box. To start a Line, type a CTRL X, then a letter L, followed by a number from 1 to 9. Position this start where you want the line to start in the document. To end the line, type CTRL X, then the letter L,followed by the number you used to start the line.

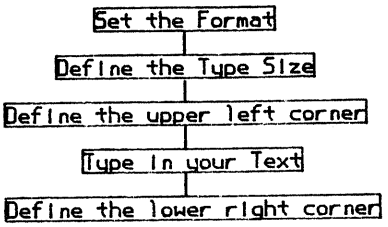
Finally, the last feature you can use to define graphics is the External Program Mode. With this command you can access the command language of the plotter and use it to produce highly customized graphic constructs. The possibilities available with this command would require another article. This command can be used to produce a circle. For example, to send a command to the plotter to plot a circle, first type in CTRL X, then the letter E followed by the command to draw a circle (ECC 200,0). When you are finished sending commands to the plotter, type CTRL X, followed by the letter E to end the command.

There are some other features that we should mention before moving on to some simple examples. One is the Dummy command. The Dummy command allows you to fill in areas of space to help you line up graphics on the page. To create dummy space type CTRL X, then the letter D, followed by a number from 1 to 9. Another feature is double-strike feature. This can come in handy when starting up a dry pen. Lastly, you can half-space and produce overstrike characters.

Now for a quick example to show you how easy it is to produce neat professional looking graphics in no time at all. First open a document on your Scripsit 2.0 or later diskette, which has the control codes copied from the Scripsit Plotter Driver diskette. Don't worry about the format or the line format, as we will set that once we are in the document.

Set the line format by typing CTRL F and edit the format line to delete the outline tab. Set the Format of the document as vertical by typing CTRL X, then F, and then V. Set the Size for 2 by typing CTRL X, then S, then 2. Now press ENTER to begin a new line. Type CTRL X, B, 1 to start the first box. Type "Set the format" as the text for the first box, then CTRL X, B, 1 to end the first box. Now press ENTER followed by

(CTRL) (L) to center the box. Now repeat this process four more times and add the text "Define the type size" to the second box and "Define the upper left corner" to the third box. Now add the text "Type in your text" to your fourth box, and finally add "Define the lower right corner" to the last box. Space down two lines and type a (CTRL) (X), (S), (4). Now type the text "How to Create a Flow Chart" and change the size to 3 by typing (CTRL) (X), (S), (3), press (ENTER) and again center the text. Go back and add lines between the boxes with the Line drawing command (CTRL) (X), (L), n where n is the number of the line.



How to Create a Flow Chart

Figure 2.

You can see how easy it would be to alter this diagram with the editing capacities of Scripsit to produce endless variations, from an organizational chart to a seating diagram for a class. For some practice, take "Figure 1" and modify it to produce a simple BASIC flowchart. Hint: you will need to use the external program function and the line command to produce flowcharting symbols.

Scripsit has many features that will allow you to masterfully come up with graphics on the spur of the moment. You will want to create sample documents that you can move, copy, and edit. Use the comment feature of Scripsit to label sample graphic symbols such as trapezoids and circles. To make a quick change on a chart, use the global search and replace feature. If you have used Scripsit User Keys before, then you know how they can simplify your work. Use them to call up external program routines for often used graphics symbols. Use the ASCII convert feature of Scripsit to combine data from programs like VisiCalc with graphics. The possibilities are limitless.

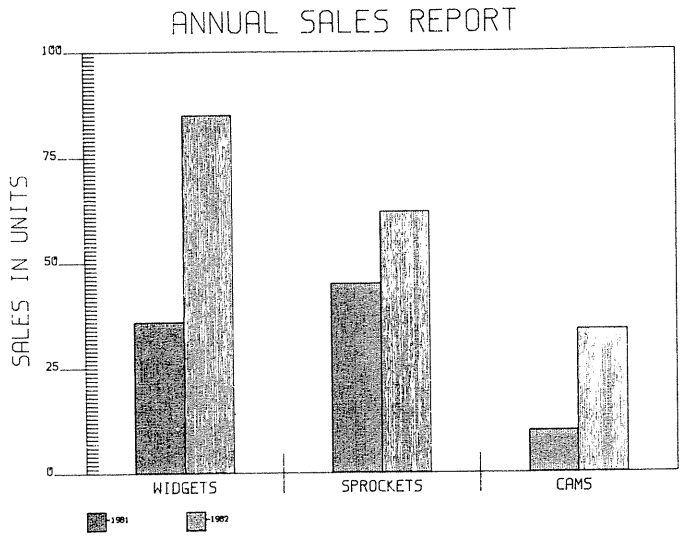


Figure 3.

As you work with the Plotter Driver package, you will see that the most difficult thing to do is to line up different size characters on the page. To aid yourself in this, remember to use the dummy command and you may want to produce a sheet of different sized characters so that you know how many it takes to fill a line. This will help with centering and lining up text of varying sizes.

For more information on producing graphics with your plotter driver, consult your multi-pen plotter manual, or check your local library and bookstore for books on graphics.

Computer Customer Service Address and Phone Numbers

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

Model I/III/4 Business Group	(817) 390-3939
Model II/12/16 Business Group	(817) 390-3935
Languages and Compilers	(817) 390-3946
Color/Model 100/Pocket Computer Group	(817) 390-3944
Hardware and Communications Group	(817) 390-2140
Educational Software	(817) 390-3302
Games, Books, and New Products	(817) 390-2133
Newsletter Subscription Problems	(817) 870-0407

Educational Computer Consortium of Ohio

ECCO, the Educational Computer consortium of Ohio, is now accepting proposals for presentation for its Third Annual Educational Computer Fair, to be held at Cleveland State University on Friday and Saturday, October 21 - 22, 1983. Classroom teachers and those with practical computer education experience are encouraged to submit proposals. We are searching for proposals in all content areas and grade levels, pre-school through college, for both beginning and advanced computer educators.

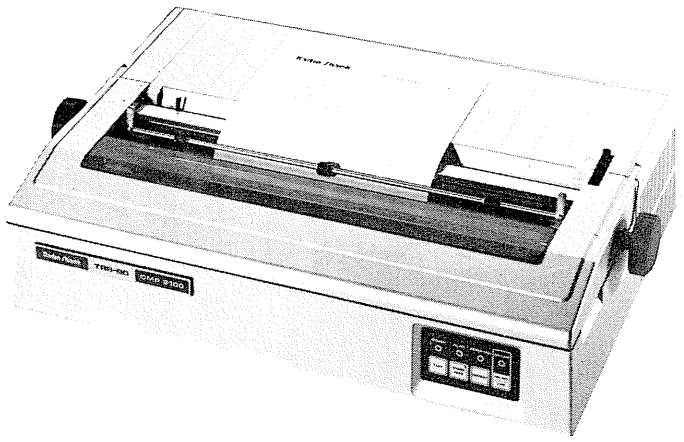
To obtain the brief proposal form, send a request to:
Ellen Richman, Coordinator
ECCO
4777 Farnhurst Road
Cleveland, Ohio 44124

The DMP-2100–Revisited

by Annette Zamberlin-Main

Well, it looks like the clandestine days of hoarding the editor's DMP-2100 are at an end. (Please see the initial article on the DMP-2100 in our March 1983 issue.) Nevertheless, before it went out of the office, I was able to become familiar with a few more of its aspects which I thought I would pass along to the readers.

The response to the DMP-2100 has been very favorable. Let's face it, the DMP-2100 is just a great printer. Since its preview in February we have read and heard of a great many positive reactions from the customers about this new printer. Particularly interesting have been some of the comments, which have appeared on various CompuServe SIGs in the past month.



If you haven't guessed it by now, the text of this article was printed from full size copy produced using the DMP-2100 and Model III SuperScripsit 1.1.0's proportional spacing capabilities. Isn't this just wonderful!!

As a point of comparison, this is a proportional print sample from a DMP-200. Proportional characters add a quality look to word processing documents.

CORRESPONDENCE QUALITY

I am still quite taken with the correspondence quality font of the DMP-2100. I referred to it in the March article, but since then have learned a little more about what goes into making each element of that great looking font of type and wish to share that information.

The correspondence quality font is available in both 10 and 12 CPI (Characters Per Inch). A comparison has been set up here between two members in the DMP series: the DMP-200 Correspondence Quality and the DMP-2100 Correspondence Quality. Each are at 10 CPI.

This is an example of the Correspondence Quality font from a DMP-200. It is produced on a 15x8 matrix at 10 CPI.

The DMP-200 is a high-density dot-matrix printer which can perform a variety of different printing operations. For instance, it can print the following types of characters:

Standard
Compressed
Condensed
Proportional
Correspondence
Graphics

This is an example of the Correspondence Quality font from a DMP-2100. It is produced on a 36x24 matrix at 10 CPI.

The DMP-2100 is a high density dot-matrix printer which can perform a variety of different printing operations. For instance, it can print the following types of characters:

Standard
Condensed
Correspondence
Proportional
Graphics

As its name implies, a dot-matrix print head forms its characters through a system of dots in rows and columns. Because of the increased number of rows and columns in the DMP-2100's matrix, each character's image is more defined than that of a character from the DMP-200 set.

The 36x24 dot matrix characters produced using the 24-wire print head on the DMP-2100

truly rival the letter quality printing capabilities of the Daisy Wheel printers.

GRAPHICS

Since this is our graphics issue, it would be appropriate here to investigate the graphics capabilities of the DMP-2100.

In the Graphics Mode of the DMP-2100, you no longer have pre-defined characters at your disposal. The responsibility for the positioning and action of the print head lies with the user.

The DMP-2100 prints two levels of graphics - a 7 bit, low resolution graphics level and a high resolution graphics level of printing. (In the March article, on page 5, two print samples appeared: one labeled "Low Resolution graphics" and other "High Resolution graphics." Both are actually examples of low resolution graphics. Sorry for the confusion.)

The regular, low resolution mode is the standard Graphics Mode that other Radio Shack printers such as the DMP-200 are capable of printing.

The high resolution mode is a special Graphics Mode that is unique to the DMP-2100. This mode takes full advantage of the printer's 24-wire print head.

For instance, a DMP-200's regular Graphics Mode creates a dot every time a wire in the print head strikes the ribbon. However, a DMP-200's print wires are much larger than the print wires in a DMP-2100's print head. For the DMP-2100 to create a dot that is equal in size to a DMP-200's dot, the print wires must strike the ribbon nine times. This creates a 3x3 dot-matrix that is equal to the single dot on the DMP-200.

To differentiate between regular and high resolution dots, we'll call the image created by a single DMP-2100 print wire striking the ribbon a "point." Nine points printed in a 3x3 matrix will create a "dot" which is the same size as the DMP-200 "dot."

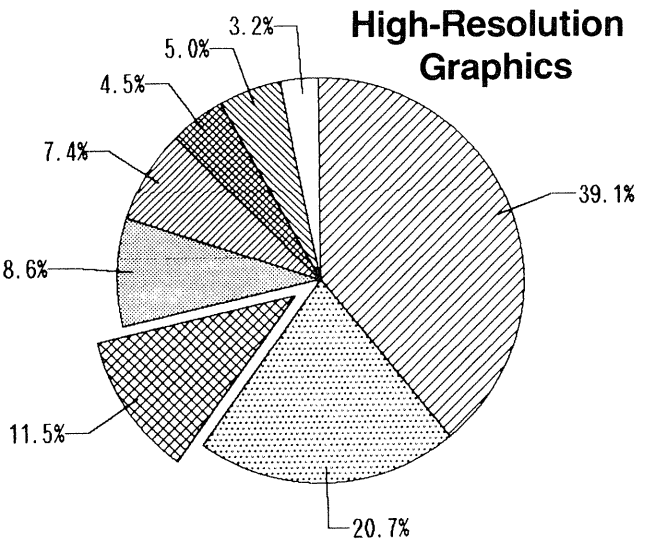
There are 816 addressable dot-columns across the 13.2x11 inch sheet of paper and 7 "up and down" addressable dots (dot rows). Which means, when all is said and done, that you can specify up to 5,712 individual dots (7x816=5,712) in each print row, in the low resolution mode.

In high resolution graphics there are 2,448 addressable point-columns "across the paper." And another 24 "up and down" addressable points (point rows). This means that there is a grand total of 58,752 (24x2448=58,752) individual points which you can specify in a single line running the width of the 13.2x11 inch sheet of paper.

Now just think about that for awhile. With the DMP-2100 you can specify up to 58,752 individual points per line in high resolution graphics. Or at 60 lines per page, that's 3,525,120 (60x58,752) addressable points on one page.

The reason the DMP-2100 has this high resolution capability is to support the high resolution characters such as the correspondence quality font we examined in an earlier section. It's going to take a tremendous amount of software support to utilize the DMP-2100's high resolution graphics. Right now, there is no commercially available software that will do this.

If you are into high resolution graphics and are writing customized software, the DMP-2100 may just be the machine for you!



ONCE MORE WITH THE PARTICULARS

The DMP-2100 (Cat. No. 26-1256), is available for $1995.00 (suggested retail price) at Radio Shack Computer Centers and participating Radio Shack stores.

Radio Shack also offers optional equipment for the DMP-2100. A Bi-Directional Tractor (Cat. No. 26-1441) for use with fanfold paper, labels, and multi-part forms, is available separately for $169.95 (suggested retail price).

A single sheet of paper can be manually fed into the printer. For larger single sheet jobs a Sheet Feeder (Cat. No. 26-1440) for automatic paper insertion and stacking is recommended. The Sheet Feeder is available separately for $995.00 (suggested retail price) at Radio Shack Computer Centers and participating Radio Shack stores and dealers.

At Home With Profile III+

The small Computer Company
P.O. Box 2910
Fort Worth, TX 76113-2910
By Ivan Sygoda, Director, Pentacle

Since most of these Profile articles have been directed toward business, I thought it would be a good time to change gears and illustrate how the program can help you enjoy leisure activities.

THE STAMP OF APPROVAL

Tens of millions of people collect things: tangible things like matchbook covers and diamonds, intangible things like bird sightings and ham radio contacts. Profile III+ is the perfect computer program to organize and collate the facts and notations relating to whatever it is you collect.

As an example, we'll set up and use a stamp catalogue file that includes the kinds of information that any collection involves—description of items, date and place of acquisition, price, condition, current value, etc. This exercise will also serve as a refresher course on defining files, screens, reports, labels, math formulas, user indexes and user menus. We'll take advantage of a Profile III+ capability we haven't used before in these articles—that of automatically extending a one-segment file over several disks as your file (and collection) grows.

COLLECTING YOUR THOUGHTS

As you'll see, Profile III+ is quite easy to use. The hard part comes before you sit down in front of the screen. You must collect your thoughts before anything else. First, think about the information you want to have on tap about your collection. Second, consider the most efficient means of entering and storing this information. Third, relax. There's no reason to be brutally efficient about any of this. After all, it's your hobby.

In my data base, which I named "STAMPCAT," I wanted to describe the configuration of each item—every type of stamp can be collected in a number of configurations such as single stamps, blocks of four, plate number blocks, first day covers, coil pairs and coil line pairs, to name some of the more standard ones. I wanted it to describe the condition of each item in considerable detail, since philatelists are somewhat fanatical about this aspect of collecting and since condition has a great effect on price and value.

I wanted my program to keep tabs on the stamp "marketplace", since stamps can also be an investment. Finally, I wanted to be able to sort my stamps according to the standard cataloguing system copyrighted by the Scott Publishing Company.

NUMBER, PLEASE

This last point proved the trickiest to deal with because of the way computers sort lists. Scott catalogue numbers are alphanumeric. Regular postal issues begin with number 1 (the first five-cent U.S. stamp of 1847). Airmail stamps begin with C1. Paid reply postal cards begin with UY1, and so on.

The problem, however, is the suffixes. Over the years, as research unearthed previously unknown printing varieties, upper-case letters were added to the catalogue numbers to wedge these new items into the numbering system. For instance, there are some varieties of two-cent stamps from 1918-20 numbered 528, 528A and 528B. But lower-case letters are also used to indicate printing peculiarities. So 528c is a "double impression" of two-cent type Va, 528A is a normal impression of two-cent type VI, and 528Ad is a double impression of type VI. (Don't laugh—you may be just as maniacal about subspecies of orioles.) The American Standard Code for Information Interchange (ASCII) sort utility used by most microcomputers will not order these catalogue numbers the way the Scott catalogue does. The catalogue order of these stamps is: 528, 528c, 528A, 528Ad, 528B, 529, but the ASCII sort order is: 528, 528A, 528Ad, 528B, 528c, 529.

The solution is to use two adjacent fields for the catalogue number. The first field (6 bytes) contains the main catalogue number, including any upper-case letter suffix. The second and following field (1 byte) contains any lower-case suffix:

Catalogue no	Suffix
528	
528	c
528A	
528A	d
528B	
529	

When I want to sort my records by catalogue number, I sort on field 1, but indicate a sort length one greater than my field length. This way, any printing varieties pertaining to a main catalogue entry will be listed properly under it.

This procedure also circumvents a feature of Profile III+ that is a convenience in most cases, but which causes problems in this application: the program ignores upper/lower-case differences when searching for matches. In the Scott catalogue system, 528b and 528B are entirely different stamps. Placing lower-case suffixes in a separate field enables Profile to distinguish between them.

THE FIELDS DEFINED

Figures 1A-1C list the fields I defined for STAMPCAT. You may notice that I didn't include a field for country, since all my stamps are U.S. If you have an international collection, you'll want to list the country first. When you sort by country, indi-

cate a field length seven greater than the country field length: country + catalogue number (6) + suffix (1). The stamps in each country will be listed in catalogue order.

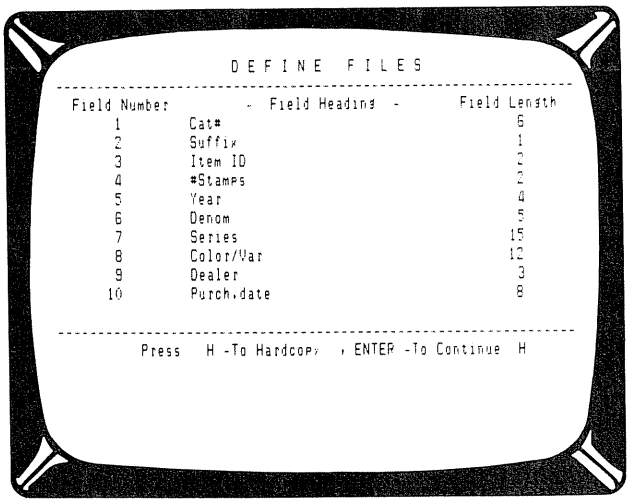


Figure 1A.

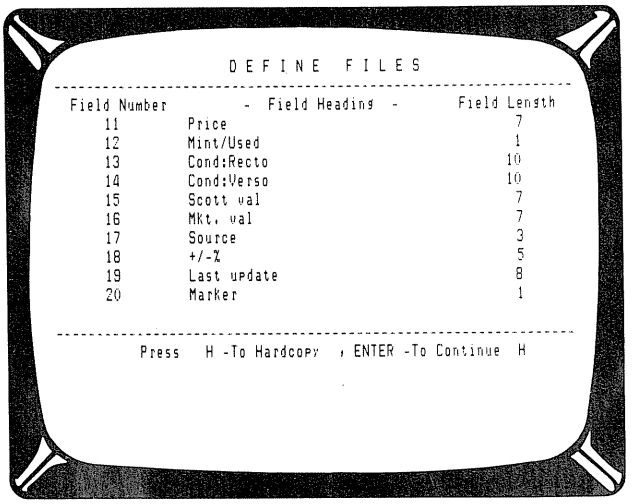


Figure 1B.

My two-character "Item ID" identifies the configuration of the item and uses the abbreviations found in many stamp auction catalogues—"BL" for a block of stamps, "PB" for a plate number block, "CS" for a coil strip and so forth. Like the Eskimos with their reputed seventeen names for kinds of snow, you can establish as many distinctions as you like, as long as you are consistent in using them. If a plate block is "PB" one day and "PL" the next, Profile will never be able to tell you what you want to know when you do an inquiry.

I've allotted two characters to indicate the number of stamps in each item because I have blocks as large as 25 stamps. Collectors love to count the number of stamps in their collection. With Profile, I can go one better. I can automatically multiply the number of stamps by their face value to get a grand total. The formula to enter into the math table is: 21 = 4*6 (the face value is equal to the number of stamps multiplied by the denomination).

Five spaces are allotted for the stamps' denomination, because the system has to cope with values from the .005 Presidential half-cent of 1938 to the $5.00 postage of the

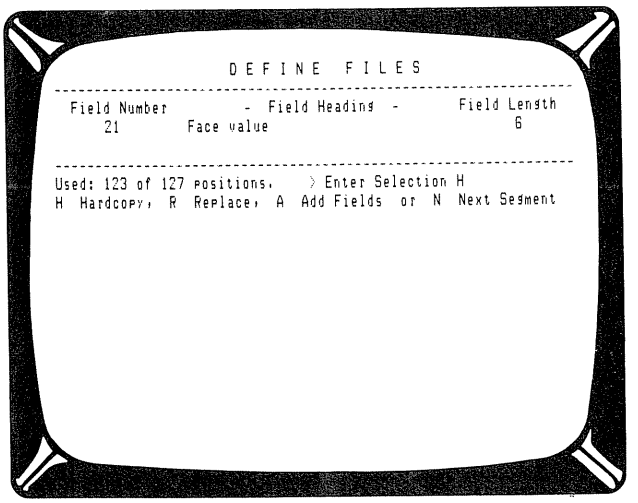


Figure 1C.

$10,000 RD185C stock transfer stamp. (A special notice to robbers: I do not have the latter stamp, so stay away.)

I chose to use a four-digit year instead of the eight-character YY/MM/DD because many interesting stamps are pre-1900. Also, the exact date of issue is unknown for many early stamps. Field 9 (Color/Variety) is the place I enter notes on what makes the stamp distinct from others of its type. Typical entries might refer to color variety, perforation count, printing process or engraving peculiarity. Under "place of purchase" I enter my three-character code for the dealer, auction house or other source. I enjoy designing codes, but if code designing doesn't thrill you, too, allot additional space for such alphanumeric entries. But before you do, I should mention that almost all the information I'm storing about each item in my collection fits on one 132-column report line, so to me it's worth the extra effort.

Since stamps have a front (recto) and a back (verso), I've allowed ten characters to describe the condition of each. This may not seem like a lot of space, but there are many standard abbreviations one can use. For instance, to grade the appearance of the back of stamps, you have "og" for original gum and "lh" light hinge mark, among others.

Field 15 contains the current Scott catalogue value, which can be useful in determining the general value of a collection for insurance and tax purposes. Many collectors receive sales and auction catalogues along with lists of prices realized. It's the philatelist's equivalent to reading the stock market quotes. I can record the latest prices for items resembling my own in field 16.

And since everyone likes to pat themselves on the back for buying low, selling high and for being generally shrewder than the next guy, I can let Profile's math package tell me how many times my stamps have appreciated in value. To do this, I subtract the original purchase from the current market value. Then I divide that result by the original purchase price and multiply by 100 to yield a percentage. The math formula is:

18 = 16-11/11*"100"I

The "I" placed in the body of the formula forces Profile to express the result as an integer.

Finally, the "marker" field is the one-character electronic "dog-ear" through which I can make special notations. For example, suppose I want to sell some stamps. After I enter an

"S" in the marker field, I can easily print a listing of those stamps I want to sell. I can also use this feature to make an up-to-date "want list." I enter the stamps I'm interested in acquiring, describe the desired condition, enter a "W" in the marker field, and print out a list for my local stamp merchant. It's impressive—he or she will know I mean business.

GETTING PERSONAL

Almost nothing is more personal than a hobby. I don't pretend for an instant that STAMPCAT will suit most other stamp collectors. Some don't care what the back of a stamp looks like. Others collect types of cancellations for which I've made no provision. Still others sell stamps as often as they buy them, and will want to record more details about these transactions. With Profile, you can have it your way. Still, whatever you collect, if there are catalogues and sales lists published, examine them for ideas about setting up your own data base—ways of organizing information, standard abbreviations and so forth.

SCREENING YOUR COLLECTION

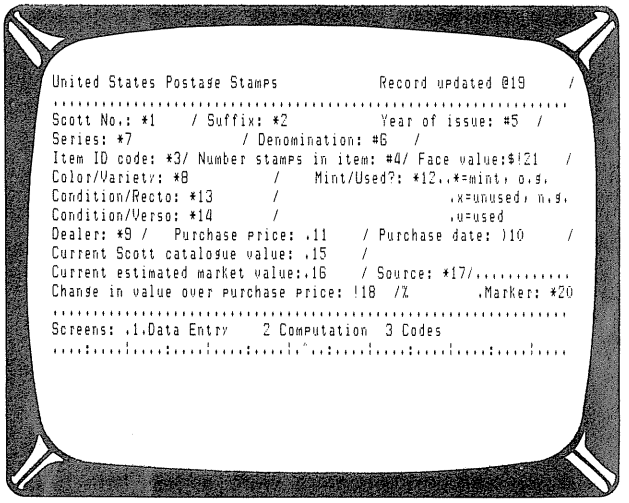


Figure 2A.

Here are two of a hundred possible screen formats for entering and reviewing my STAMPCAT data fields. Figure 2A shows the way STAMPCAT/PM1 (Screen format no. 1, my data entry screen) looks when I format it from the Creation menu. I use this screen to enter data. Figure 2B shows the way it looks during typical use. I often reserve a screen for the original entry of information into a record.

Once the data is in the file, the data entry prompts become superfluous, and so I eliminate them from the screens I work with afterwards. Figures 3A and B show such a working screen, STAMPCAT/PM2, which I use to enter and update marketplace information. I use "!" to protect the information in most of the fields on my screen format no. 2 (figure 3A).

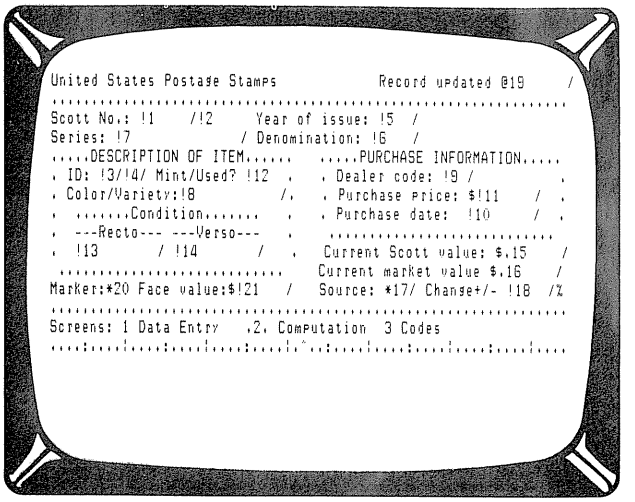


Figure 3A.

Profile III+ lets you access the complete collection of Model III special characters to make attractive screen layouts. It's easy. At any time during screen creation, press SHIFT and @ together when the cursor is positioned where you want a special character to appear on the screen. The cursor jumps to the two bottom lines, where the special characters are arrayed. Position the cursor over the desired character and press ENTER. Presto, you're an artist. Experiment.

Here's a trick: If you're worried about forgetting the codes and abbreviations you make a part of your application, list them all on a screen you can call at the touch of a button.

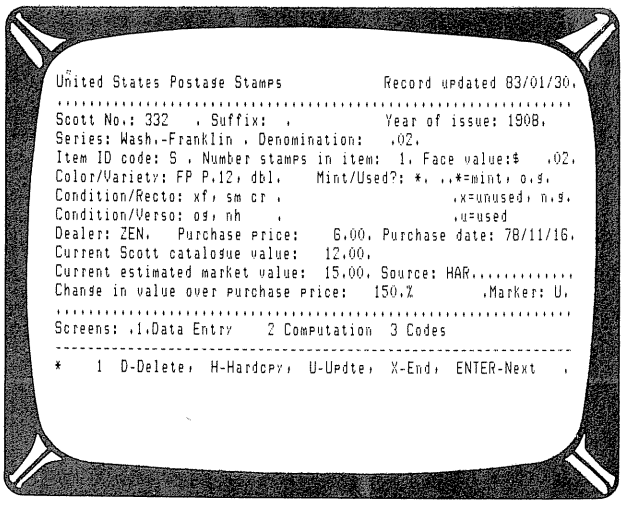


Figure 2B.

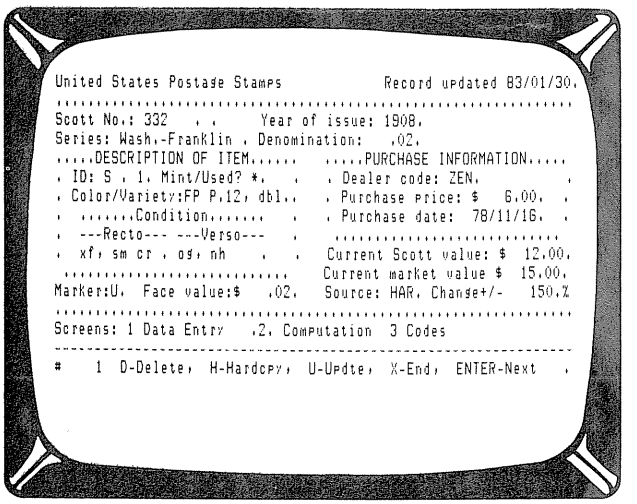


Figure 3B.

There's one catch. If you list codes only, you'll get an error message stating that there are no fields in your format. So stick a field in the bottom corner of the screen, and Profile will keep its mouth shut. I used field 2 since it's short, often void and can be formatted with just one digit. Figure 4A shows the way STAMPCAT/PM3 (Screen 3) looks when formatted. Figure 4B shows the way it looks during use. The "phantom" field is on the bottom right.

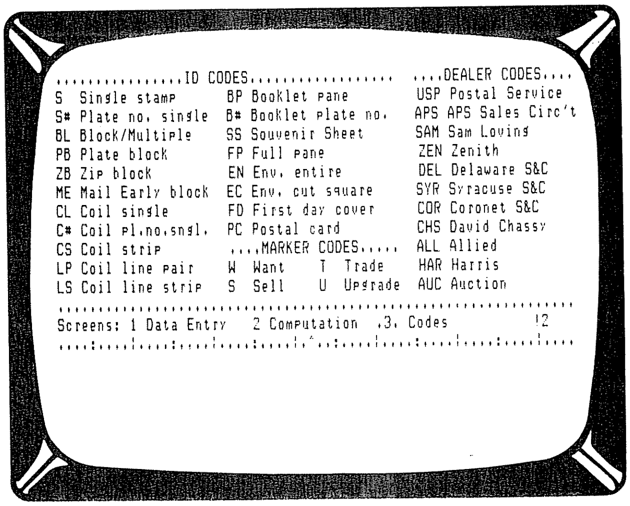


Figure 4A.

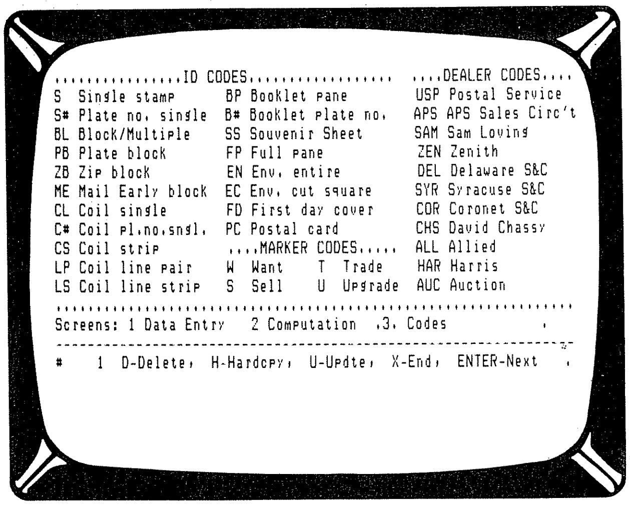


Figure 4B.

REPORTS AND LABELS

```
                                                                 *                         1         1         1         1
         1         2         3         4         5         6    *** 7         8         9         0         1         2         3
----+----0----+----0----+----0----+----0----+----0----+----0----+*---0----+----0----+----0----+----0----+----0----+----0----+----0--
......................T.I.T.L.E.-.L.I.N.E.S....................
                                                            UNITED STATES POSTAGE STAMPS
                                                              Ivan Sygoda, New York NY
                                                              Date: /        Page #
...................H.E.A.D.I.N.G.-.L.I.N.E.S...................
Sort criterion:?           ===Description==== Item  Face  ========Condition======== Purchase Information ======Current value======
Scott no. ====Series===== Denom =Color/Var.= ID #  value */u ==Recto=== ==Verso=== DLR ==Date== =Price= =Scott= Market= Scr  +/-%
......................F.I.E.L.D.-.L.I.N.E.S....................
*1    /*2 *7          /*6   /*8         /*3/=4 =21   /*12*13     /*14        /*9  /*10    /=11    / =15    / =16    /*17/*18 /
```

Figure 5A.

```
                                                            UNITED STATES POSTAGE STAMPS
                                                              Ivan Sygoda, New York NY
                                                              Date: 05/02/83 Page    1


Sort criterium:            ===Description==== Item  Face  ========Condition======== Purchase Information  ======Current value=======
Scott no. ====Series===== Denom =Color/Var.= ID #  value */u ==Recto=== ==Verso=== DLR ==Date== =Price=  =Scott= Market= Src  +/-%

332       Wash.-Franklin  .02 FP P.12, dbl S   1    .02 * xf, sm cr   og, nh     ZEN 78/11/16    6.00     7.50   32.00 HAR  433
334       Wash.-Franklin  .04 FP, P.12 dbl S   1    .04 * xf gem      og, nh     ZEN 78/11/16   75.00    27.50   85.00 HAR   13
335       Wash.-Franklin  .05 FP, P.12 dbl S   1    .05 * vf-xf       vlh        ZEN 78/11/16   65.00    40.00  105.00 HAR   62
336       Wash.-Franklin  .06 FP, P.12 dbl S   1    .06 * f-vf        nh         SAM 74/06/05    8.00    50.00  120.00 HAR 1400

                                               4    .17                                        154.00   125.00  342.00



RECORDS SELECTED     4
```

Figure 5B.

Figure 5A shows report format 1 (STAMPCAT/PR1) and figure 5B shows a sample print-out. Figures 6A and B show STAMPCAT/PR2, which I use as a want list. Figure 7A shows a possible label format (STAMPCAT/LB1), and 7B illustrates how it turns out using my printer's proportional-space correspondence-quality font. The labels are great, especially if you

```
                                                                *
         1          2          3          4          5          6      *** 7          8
----+----0----+----0----+----0----+----0----+----0----+----0----+----0----+*---0----+----0
.....................T.I.T.L.E.-.L.I.N.E.S.....................
                     UNITED STATES POSTAGE STAMPS: WANT LIST
                            Ivan Sygoda, New York, NY
                               /        page #    /
.....................H.E.A.D.I.N.G.-.L.I.N.E.S.....................
==SCOTT==  ==============DESCRIPTION==============  =========CONDITION=========
=Cat.no.=  ====Series===== Denom Color/Var.   ID #   */u ===Recto=== ===Verso===
.....................F.I.E.L.D.-.L.I.N.E.S.....................
*1    /*2  *7              /*6   /*8          /*3/*4  *12 *13       / *14       /
```

Figure 6A.

```
                     UNITED STATES POSTAGE STAMPS: WANT LIST
                            Ivan Sygoda, New York, NY
                               05/02/83 page    1


==SCOTT==  ==============DESCRIPTION==============  =========CONDITION=========
=Cat.no.=  ====Series===== Denom Color/Var.   ID #   */u ===Recto=== ===Verso===

332        Wash.-Franklin    .02 FP P.12, dbl S   1  *   xf, sm cr     og, nh
334        Wash.-Franklin    .04 FP, P.12 dbl S   1  *   xf gem        og, nh
335        Wash.-Franklin    .05 FP, P.12 dbl S   1  *   vf-xf         vlh
336        Wash.-Franklin    .06 FP, P.12 dbl S   1  *   f-vf          nh



     RECORDS SELECTED      4
```

Figure 6B.

like to make your own album instead of placing your stamps in someone else's squares. Note: If you want to use proportional spacing, you have to format "one-wide" labels because the left margin of the right hand columns of labels will be ragged. Print half, then turn the sheet of labels upside down and feed them in this way.

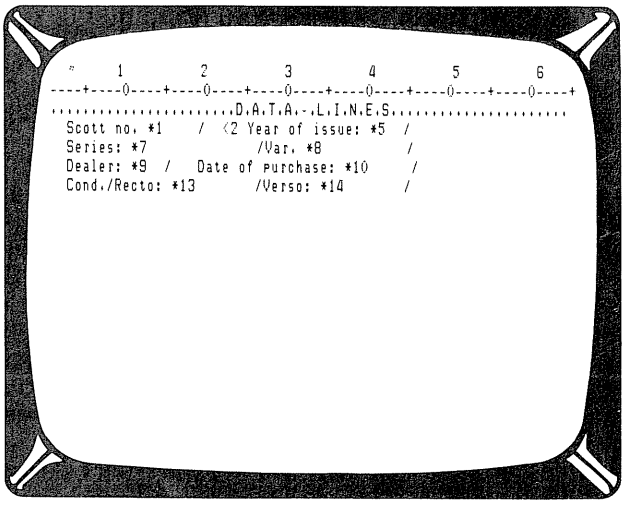


Figure 7A.

```
Scott no. 332       Year of issue: 1908
Series: Wash.-Franklin  Var: FP P.12, dbl
Dealer: ZEN   Date of purchase: 78/11/16
Cond./Recto: xf, sm cr  Verso: og, nh

Scott no. 334       Year of issue: 1908
Series: Wash.-Franklin  Var: FP, P.12 dbl
Dealer: ZEN   Date of purchase: 78/11/16
Cond./Recto: xf gem     Verso: og, nh

Scott no. 335       Year of issue: 1908
Series: Wash.-Franklin  Var: FP, P.12 dbl
Dealer: ZEN   Date of purchase: 78/11/16
Cond./Recto: vf-xf      Verso: vlh

Scott no. 336       Year of issue: 1908
Series: Wash.-Franklin  Var: FP, P.12 dbl
Dealer: SAM   Date of purchase: 74/06/05
Cond./Recto: f-vf       Verso: nh
```

Figure 7B.

USING AN INDEX

When you first enter data in your file, accessing it is reasonably quick and efficient. You probably don't have many records. The ones you do have are most likely in the order of your original manual filing system. You have only a few (if any) deletions. All this helps to facilitate the speed with which Profile searches through your records to find those you want. Calling a record by its record number is the fastest way to get to it, but unless you have an amazing memory, you probably don't remember which number goes with which item in your file. You could always print a report that includes the record number of each of your records, using the record numbers as a "home-brew" cataloguing system for your collection. But then you would need to keep that report handy all the time if you want Profile to find a particular item in your file as quickly as possible. Since this is rather inconvenient, Profile lets you set up an index to speed things up. If you have a ready-made cataloguing system, like that of the Scott catalogue for stamp collectors, you can set up an index based on it, which will be a big help in finding items in your file.

Whenever I want to see one of my stamps, I enter its Scott catalogue number, not its record number.

Here's a schematic illustration of how an index works. My records might originally have been entered in the following (random) order:

Rec.no	Cat.no	Suffix
1	C15	
2	1619	a
3	100A	
4	C2	
5	20	a

I need to put this file in catalogue number order, so I sort by field 1 (length = 7). Profile III + constructs an inquiry index called STAMPCAT/IX1 which looks something like this:

Cat.no	Suffix	Rec. No.
20	a	5
100A		3
619	a	2
C2		4
C15		1

This index file takes up very little room on the disk. Each record in it is only 10 bytes long (the sort field length of 7 plus 3 bytes for housekeeping, i.e., the "pointer") as opposed to the 123 bytes for the full record. Since the index is in the order I want to access my files, it finds records fast. And it's there waiting for me whenever I turn on the system and enter STAMPCAT. However, there's one catch: Whenever I add or delete records, I have to rebuild this index so that it will incorporate the changes. As we'll see below, Profile can be set up to accomplish this automatically.

BUILDING AN INDEX

Build Index is selection 2 on the Profile Runtime menu. It only takes a few keystrokes to build an index. Make a note of each one, because you'll need to enter them in the BUILD file which automates the procedure. Following are the prompts I encounter and the responses I should enter to build my catalogue number index:

PROMPT	RESPONSE
Enter File Name	STAMPCAT1
Enter Field Number to Sort	1
Enter Length For Sorting or Press ENTER for Length of Field	7
Capacity 3756 Records	
Enter Selection Field Number or Press ENTER to Select All Records	ENTER

The program then tells me how many records were selected and returns me to the Runtime menu. When I examine the DIRectory, I find that Profile has now created STAMPCAT/IX1 with a "Logical Record Length" of 10. Also, there is a new prompt for Inquire, Update, Add. Passing over "Enter Record Number" by pressing ENTER, I find: "Index by: Cat#" with the cursor blinking for a response. "Cat#" is the heading I used for field 1 when I defined files. Pressing ENTER again calls up the sort and select options. There are two ways to answer the index prompt. I can enter a specific catalogue number, in which case the program searches for it in the new index and then displays that record on the screen. Or I can enter the "wild card" symbol, " = ". This brings the first or lowest catalogue number to the screen. (Technically, Profile fetches the first "logical" record as opposed to the first "physical" record.) I can then use the up and down arrows to examine my records in catalogue number order.

AUTOMATION

Since the index must be rebuilt every time you add or delete records, it's convenient to do this with just one keystroke from a user menu. The process involves BUILDing a DO file that then becomes a menu choice. From TRSDOS Ready, I type "BUILD REBUILD," since I decided to call my DO file "REBUILD." In response to the prompt, I enter "EFC8 (STAMPCAT,1,Rebuild CatalogueNumber Index)."

"EFC8/CMD" is the Profile Runtime program which builds the index. "STAMPCAT" is the file name. "1" refers to a screen, and, though it's irrelevant here, it's important for other operations, and so it is part of the format. "Rebuild Catalog Number appears on the screen when Profile executes REBUILD. TRSDOS again prompts for an entry, so I enter "1," the response to "Field Number to Sort By." The next entry is "7," the desired sort field length. The final entry is simply ENTER, because I wish to "Select All Records." I press BREAK to complete the process. When I now examine the DIRectory, I find REBUILD/BLD and notice that it occupies one gran. To test the new BUILD file, I enter "DO REBUILD" at TRSDOS Ready.

USER FRIENDLIER

The final touch is the creation of a customized user menu. With a little bit of planning, you can partially or completely automate the procedures you use most often. I call my menu STAMPCAT, which the directory lists as STAMPCAT/CMD. Figure 8 shows how my menu creation screen looks when I finish defining my choices. Selection 7 is the BUILD file we just created to rebuild my catalogue number index. Selection 2 takes me right up to the "Enter Record Number" prompt in Inquire, Update, Add. Selection 1 calls ADDSTAMP/BLD, which displays the first blank record on the screen, ready to accept a new stamp.

ALLSCAN/BLD calls screen 2 and enters the wild card symbol " = " in answer to the Cat# prompt. Finally, selection 8 takes me back to the Profile Runtime menu (RM). (Keep in

```
1.        Add a stamp to the collection
DO ADDSTAMP
2.        See a particular stamp
EFC9 (STAMPCAT,2,See a particular stamp)
3.        Scan the entire collection
DO ALLSCAN
4.        Print collection report
EFC8 A(STAMPCAT,1,Print collection report)
5.        Print want list
EFC8 A(STAMPCAT,2,Print want list)
6.        Print album labels
EFC8 B(STAMPCAT,1,Print album labels)
7.        Rebuild catalogue number index
DO REBUILD
8.        Call Profile III+ Runtime menu
RM
```

Figure 8.

mind that indexing adds another (ENTER) to a "Scan Records" or "Add Records" entry into the file.)

IT JUST GROWS, LIKE TOPSY

I said at the beginning that STAMPCAT takes advantage of a special Profile capability. When my collection grows to the point that this one-segment-only file completely fills the available space on my Runtime diskette, the segment automatically jumps to the formatted data diskette I have waiting for it in the next drive. And when that drive is filled, it extends to drive 2, and then, finally, drive 3. STAMPCAT/KX1, KX2, and KX3 are the results of this utility. Rather neat, eh? All I need is enough stamps. Happy collecting!

COMING ATTRACTIONS

Next month we'll discuss how to access your Profile data files from BASIC. We've received many inquiries on that subject. In future issues we'll introduce the fantastic Hard Disk version of Profile III + (26-1593), which should be available about the time you read this. This enhanced Profile incorporates the advanced sort and select features Model II users know as Prosort. Then we'll examine the ways your Profile data bases can interface with Scripsit and VisiCalc. Let us know if there's anything else you'd like to read about. If you use Profile in novel applications or would like to share tricks and techniques you may have discovered, please write and describe them to us. You, too, can be in print.

PROFILE Editor's Note: This is Mr. Sygoda's eighth article in a series of 'how-to' Profile III + articles. Other articles in the series will be published over the next few issues in this column. We hope that you enjoy this feature, and we look forward to your comments and questions on Profile III+.

Pentacle is a New York City-based non-profit service organization specializing in administrative services for performing art groups.

High Motivation Reading from Radio Shack

Radio Shack's High Motivation Reading Series packages, available in Model III and Model I versions, help build reading skills at the critical fourth through sixth grade reading levels. The series is designed for classroom use as a supplement to the regular reading curriculum. The motivational design of each package makes HMRS ideal for enrichment and/or remedial use with students at a wide range of grade levels.

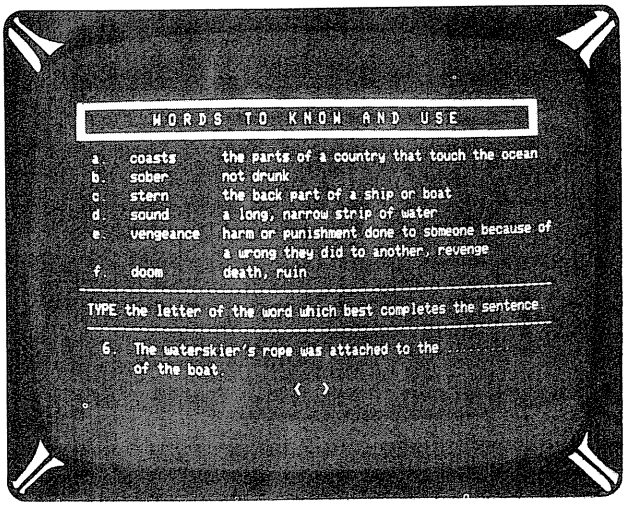
Each HMRS package contains four illustrated-format reading books, one read-along audio cassette tape, and a diskette with reading comprehension activities to be completed at the computer. Additional activities are in the teacher's manual for each package. Model III versions of HMRS also include repeatable spelling and vocabulary exercises on the activities diskette. To use the HMRS packages, you also need the Model III or Model I version of TRS-80 MicroPILOT™ (Cat. No. 26-2718 or 26-2205).

HMRS packages currently available for the TRS-80 Model III 32K disk system are: *The Hound of the Baskervilles, Dracula, Moby Dick, The Beatles, Charles Lindbergh/Amelia Earhart* (the Lindbergh/Earhart package contains two activities diskettes), and *20,000 Leagues Under the Sea*. The same titles are available for a 32K Model II disk system with lower-case driver, although quantities available may be limited. *Dracula* and *Moby Dick* are currently available for the Model I 32K disk system with lower-case driver.

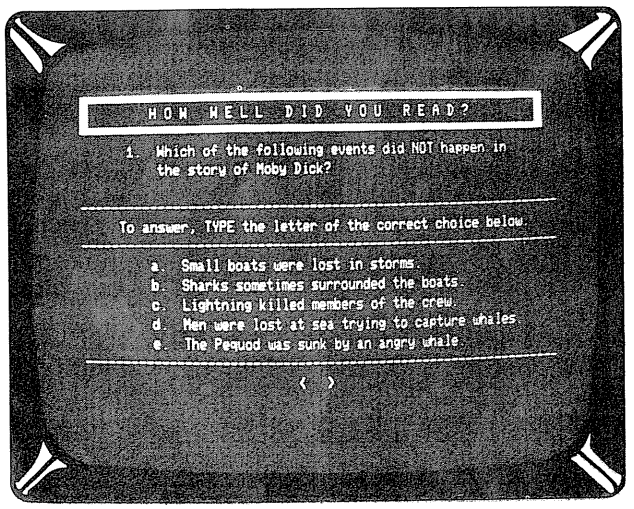
The reading book in each package condenses the classic story (or the biography), retelling it in words appropriate to the reading level. The illustrated format of the books helps make the stories accessible, and makes the reading process less threatening and more enjoyable. HMRS helps students build reading skills while gaining exposure to stories they

might not otherwise read, or might not read until they were much older. Once they have enjoyed the HMRS version of a book like *Moby Dick* or *The Hound of the Baskervilles,* students may well want to tackle the original!

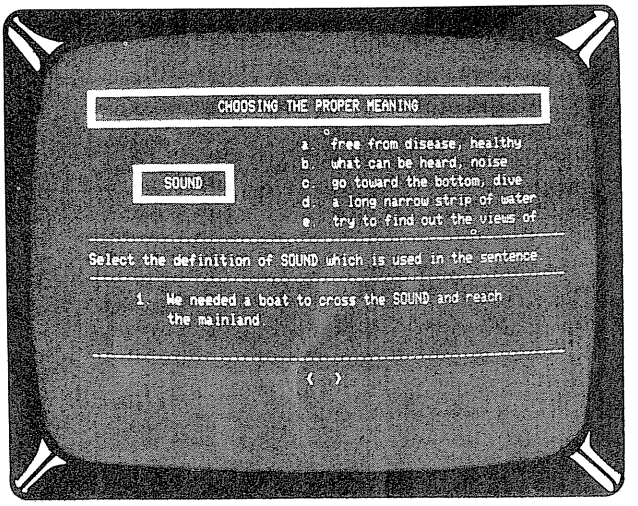
The exercises on diskette vary from title to title, but let's take a look at a typical set from *Moby Dick*. Both the Model III and Model I versions of *Moby Dick* have three sets of reading comprehension exercises: "Words to Know and Use," "How Well Did You Read?," and "Choose the Proper Meaning." "Words to Know and Use" presents a list of words and defines the words as they are used in the story. Then, the student is given a series of sentences and is asked to fill in the blank with the correct word.



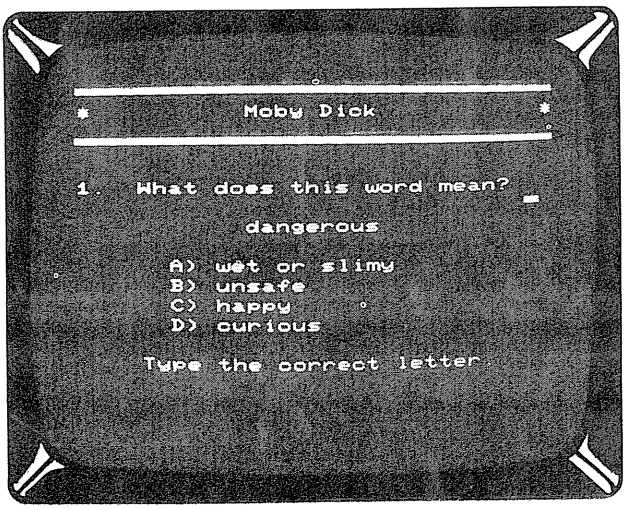
When the student responds correctly to an HMRS exercise, he or she sees a positive feedback message and then gets another sentence. For an incorrect response, the student is prompted to try again. A new sentence isn't presented until the student enters the correct response. At the end of each program, the student's score on the activity is displayed. (An optional HMRS Student Records System is also available separately in Model III and Model I versions for teachers who want to save HMRS student scores on diskette.) "How Well Did You Read?" tests the student's reading comprehension and recall of details by asking questions like this:



"Choosing the Proper Meaning" asks the student to tell what meaning of a word is used in each of a set of sentences. The exercise below is from a sequence that presents several sentences using "SOUND" and requires the student to choose a different definition of "SOUND" for each sentence.



The Model III version of *Moby Dick* also has Spelling and Vocabulary exercises that emphasize words used in the story. Spelling exercises display a sentence, highlight a word in the sentence, and then make the highlighted word disappear. To work the exercise, the student types the correct spelling of the word that disappeared. In the Vocabulary exercises, a word and a list of definition choices are displayed on the screen. The student then types the letter that identifies the correct definition.



HMRS programs can be used with students at a wide variety of grade levels and skill levels. Students who are reading below grade level should find the illustrated format of the readers and the dramatic presentation of the story on audio tape to be aids in increasing comprehension and vocabulary skills. The series may provide younger students reading above grade level with an added challenge, and may

(Continued on page 17)

The New TRS-80 Micro Color Computer

by Linda Miller

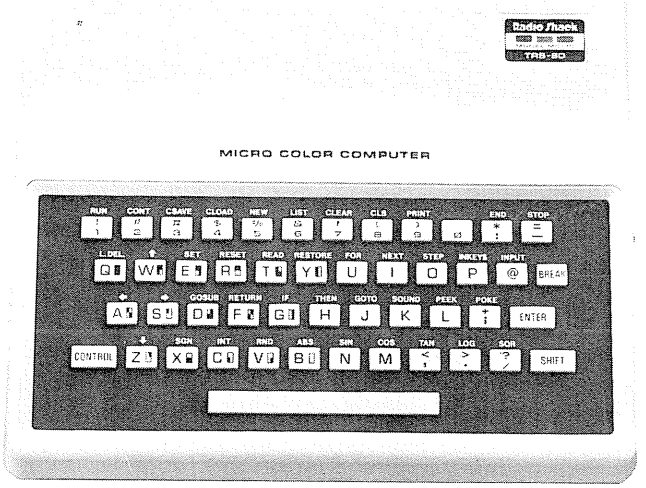
The TRS-80 Micro Color Computer Model MC-10 offers graphics capability and BASIC programming in a compact, inexpensive unit. Some outstanding features of the MC-10 are:

- A keyboard with keys that move when pressed. This is not a membrane keyboard.
- A built-in RS-232 serial port for communications or for connecting other serial devices such as serial printers or plotters.
- Keyword input using two simple keystrokes.
- Low resolution graphics in a 64 × 32 matrix.
- Sound. Create your own music or sound effects.
- Microsoft Color BASIC

The MC-10 is approximately 8 1/2" wide by 7" deep by 2" high. It has 48 keys and can be connected to a TV (preferably color) via its own antenna switch. A power supply and a well written, easy to understand manual are also supplied with the Micro Color Computer. The suggested retail price for the computer is $119.95.

THE KEYBOARD

The MC-10 has a "QWERTY" keyboard with keys that move (travel) when pressed. Above thirty-seven of the keys are familiar BASIC words and functions. Many keys have two characters on the key cap much like a typewriter keyboard. Graphic characters are located on the alphabetic keys Q, W, E, R, T, Y, A, S, D, F, G, Z, X, C, or V. The second characters which appear above or to the right side of the primary character are created by pressing [SHIFT] and the character key. To create the graphic character on the A key, press [SHIFT] [A] and the graphic character ▚ appears on the screen.



Above other keys are commands (L. DEL, arrow keys) and BASIC words. The commands and BASIC words can be duplicated by pressing the [CONTROL] key and the key below the desired word. To enter RESTORE from the keyboard, press [CONTROL] [Y] and RESTORE appears on the screen. Keyword input in this manner makes for faster program entry.

Special instructions are located on the Q,A,W, S, and Z keys. [CONTROL] [A] is the backspace instruction; [CONTROL] [Q] deletes the entire current program line, [CONTROL] [W] generates an up arrow which is the exponentiation operator, and [CONTROL] [S], [→], and [CONTROL] [Z] [↓] are reserved for special purposes.

[SHIFT] [0] toggles the computer back and forth between upper and lower case. The lower case characters appear as reverse-video, upper-case characters on the screen but print on paper in lower case.

THE MC-10 HAS CONNECTIONS AND MEMORY

On the back of the MC-10 are a cassette recorder connector, a slot for future memory expansion kits, a RESET button, and an RS-232 connector so the MC-10 can be used with serial devices including modems and Radio Shack's serial printers or plotters.

The MC-10 can store over 3K bytes in RAM for user access at a single time. When the power is turned off, the memory is erased. For permanent storage, programs and data must be written to cassette tape.

ERROR MESSAGES

The MC-10 displays a full range of error messages—19 in all—in response to programming mistakes.

ADD A LITTLE COLOR TO YOUR LIFE

There are nine distinct colors that can be generated with the MC-10.

Color	Code
Black	0
Green	1
Yellow	2
Blue	3
Red	4
Buff	5
Cyan	6
Magenta	7
Orange	8

Each of the 16 graphic characters may be displayed with a different background color. When [CONTROL] [0] is pressed, the cursor changes colors, and the new color be-

comes the background color for the next graphic character. Printing CHR$ (128) through 255 will display the 16 character graphic characters in all eight possible colors. Graphic characters can also be entered directly from the keyboard. Pressing (SHIFT) and the key with the selected graphic character, displays the character on the screen. In program lines, the graphic character is enclosed in quotation marks.

```
10 PRINT"■"
```

DO I HEAR MUSIC?

The Micro Color Computer is able to control and program the sound which comes from the TV's speaker. Volume is adjusted on the TV, and pitch and duration are controlled with the BASIC SOUND statement:

SOUND pitch-code, duration (ENTER)

Pitch-code can be any number from 1 to 255 with 1 being the lowest and 255 the highest pitch. Duration is any number from 1 to 255, and duration is specified in units of .06 seconds. SOUND 39,10 produces middle-C for .6 of a second.

ARRAYS

The MC-10 supports both single and double dimensioned arrays. Arrays that have more than 10 elements need to be dimensioned using the DIM statement.

USING A CASSETTE

The cassette baud rate is 1500. This is a transfer rate of approximately 190 characters per second or 11,000 characters per minute.

Programs can be saved to (CSAVE) or loaded from (CLOAD) cassette. The contents of an array are saved to cassette using CSAVE* and loaded back into memory from cassette using CLOAD*.

SKIPF searches for a program previously saved to tape.

THE MC-10 AS A CALCULATOR

Like other computers, the MC-10 can be used as a number cruncher. It adds (+), subtracts (−), multiplies (×), and divides (/) and performs mathematic functions.

RELATIONAL AND LOGICAL OPERATORS

The relational operators are:

Operator	Function
>	Is greater than
<	Is less than
=	Is equal to
< >	Is not equal to
< =	Is less than or equal to
> =	Is greater than or equal to

Logical operators test values to see if they are true (on) or false (off). The logical operators AND, OR, and NOT can be used in conjunction with IF . . . THEN to perform true/false tests.

STRING AND NUMERIC VARIABLES

The MC-10 handles two types of variables: numeric and string. Numeric variables are designated by any two character combination as long as the first character is a letter and the second character is a letter or a number.

String variables are formed the same way as numeric variables with the addition of a dollar ($) sign to indicate a string variable. String variables can be added (concatenated) together as in the example below.

```
NAME$ = FIRST$ + MIDDLE$ + LAST$.
```

THE MICRO COLOR BASIC DIALECT

BASIC IN THE MICRO COLOR COMPUTER		
ABS	IF/THEN	PRINT TAB
ASC	INKEY$	PRINT@
ATN	INPUT	READ
CHR$( )	LEFT$	REM
CLEAR	LEN	RESET
CLOAD	LET	RESTORE
CLOAD*	LIST	RIGHT$
CLS	LLIST	RND
CONT	LOG	RUN
COS	LPRINT	SET
CSAVE	MEM	SGN
CSAVE*	MID$	SIN
DATA	NEW	SKIPF
DIM	ON/GOSUB/RETURN	SOUND
END	ON/GOTO	SQR
EXP	PEEK	STOP
FOR/TO/STEP NEXT	POINT	STR$
GOSUB/RETURN	POKE†	TAN
GOTO	PRINT	VAL

†POKE may be used for addresses in video RAM only.

YOU SAY YOU WANT FRIENDLY?

The Micro Color Computer (Model MC-10) has color, graphics, sound, a real keyboard—in short, a lot for its remarkably low price. The combination of affordability, power, and easily understood documentation make this unique computer friendly to all including the first time user.

HMRS (From page 15)

result in an increased enjoyment of reading. Older students who do not enjoy reading can become involved in the reading process through the motivational format, and develop an interest in, and positive attitude toward, reading. For students reading at grade level, this series provides an exciting and different way to sharpen reading skills.

Radio Shack's HMRS packages are available from your local Radio Shack store or Computer Center, ON SALE THROUGH SEPTEMBER 30, 1983. Model III versions are sale priced at $41.97 each (reg. $69.95), except for *Charles Lindbergh/Amelia Earhart,* which is sale priced at $44.97 (reg. $74.95). The Model I packages are sale priced at $23.98 (regularly $59.95) with *Charles Lindbergh/Amelia Earhart* at $25.98 (regularly $64.95). The optional HMRS Student Records System for Model III (26-2521) is on sale for $17.97; sale price for the Model I version (26-2508) is $9.98. Sale prices are suggested and may vary at individual stores and dealers. TRS-80 MicroPILOT, which is required to run HMRS programs, is itself a complete command-oriented courseware authoring system that you can use to create your own computer assisted instruction programs. Model III MicroPILOT (Cat. No. 26-2718) is on sale for $71.97 (regularly $119.95). The Model I version (Cat. No. 26-2205) is on sale for $39.98 (regularly $99.95). For more information on how you can use the High Motivation Reading Series, contact your local Radio Shack Store or Computer Center, or contact the Radio Shack Regional Educational Coordinator for your area.

Charting—An Important Electronic Marketing Tool for Agriculture

by Kathryn Frame

Charts—a visual picture of exactly what is happening in the agricultural marketplace—are now available to farmers, ranchers, ag bankers and agribusinessmen via Agri-Star, America's Agricultural Information and Computing Network.

This new electronic marketing tool can aid those in agriculture to predict future price trends of crops and livestock. This, in turn, can improve the timing of sales and purchases, so important to a profitable business.

John Holst, a Center Junction, Iowa, farmer, uses bar charts and moving average charts to determine his marketing decision—when to buy and when to sell corn and soybeans.

The Agricultural Services Department of Marine First National Bank, Janesville, Wisconsin, pulls charts on a weekly basis to advise its customers as to when to buy and/or sell their crops and livestock. According to James E. Atchinson, Assistant Vice President, charts pulled on a weekly basis by the department include corn, soybean, fat cattle and hogs.

Once a microcomputer or data terminal, a modem, and a printer have been added to the list of hardware, an agricultural businessman should have to insure a maximum profit from his product, Agri-Star can provide:

- Information at much less cost than printed market advisory services.
- Only the information desired, eliminating hours of research and reading time;
- Information continuously updated 24-hours-a-day through AgriScan—a service of Agri-Star's continuously updated agricultural business information analysis and recommendations.
- In addition to charts, AgriScan also offers business: markets, prices, finance, weather, market analysis, market advisories, and news.
- Codefile—an economical and time-saving function which allows a user to build an online file of up to 10 report access code numbers he regularly uses. And, the user can store as many codefiles as he needs. These access codes are then stored in the Agri-Star computer for use anytime he wants to connect to Agri-Star.
- Rapid use of a printer, and, thus low connect time, is also possible, with the use of a RUN function in tandem with the codefile function.

WHAT CHARTS ARE CURRENTLY AVAILABLE ON AGRI-STAR'S AGRISCAN?

There are five types of charts on AgriScan for use in analyzing market trends and changes in price direction:

- Buy/Sell Decisions: Bar Charts
- Spot Trends: Moving Averages
- Overbought/Oversold: Relative Strength Index
- Measure of Trading Activity: Volume Charts
- Market Change: Open Interest
- Spot Signals: Point and Figure (to be available in the near future)

Commodities charted on AgriScan include: corn, T-Bills, wheat, soybeans, soymeal, cotton, feeder cattle, live cattle, hogs, and pork bellies.

Both Top Farmer Current Charts (TFC) and Top Farmer Historical (TFH) are available for each of the five charts.

TOP FARMER CURRENT CHARTS (TFC):

Using Agri-Star's keyword search ability, the chart of current market activity desired can be obtained by typing in the information desired. This will yield a menu of Top Farmer Current (TFC) charts. The charts pulled up on the screen will be those of current market activity. The computer will then assemble a menu of charts covering the most recent two months of trading activity for any particular chart type and contract month chosen.

The menu seen will show all charts available for that chart type and contract month, listed by the contract months covered in the AgriScan charting service.

(Once the chart access code is entered, the printer will produce a continuous printout of the chart displayed with the Y axis shown horizontally, and the X axis shown vertically. This orientation is exactly the opposite of how the X and Y axes normally appear and has been adopted for AgriScan charts in order to accommodate the 80-character limitation of the majority of printers. Once the chart copy is printed and turned on its side, a complete chart showing the latest two months trading for the commodity and contract being tracked is ready for analysis.)

If "background" chart activity on a particular TFC chart is desired, a chart of previous market activity can be obtained by accessing the Top Farmer Historical (TFH) charts in AgriStar.

TOP FARMER HISTORICAL CHARTS:

"TFH" charts cover market activity for all contracts and all chart types represented in TFC charts, but with one major difference: TFH charts in AgriScan end where TFC charts

begin. That is, TFH charts cover a maximum of 16 months of market activity, counting backwards in time from week one of the currently recorded TFC chart.

BUY/SELL DECISIONS: BAR CHARTS:

Bar charts in Agri-Star give a graphic representation of each day's trading activity for a specific commodity and contract month. The "picture" that bar charts give of price movements can be used to anticipate future market trends and changes in price direction.

These charts are: updated each day shortly after the markets close; are plotted from the daily high, low, and closing prices of a specific commodity contract; and are plotted Monday through Friday only, with holidays left blank.

By use of technical bar chart analysis, current short and long term trends can be determined. Once this determination is made, major marketing decisions can be made—when the major trend is up, there would be little reason to rush sales of farm products (unless a reasonable profit can be locked in, and, by doing so, avoiding future price risk). When the major trend shifts downward, forward pricing is usually immediately available.

In following trends indicated by the consistent use of bar charts in conjunction with the information surrounding the Payment-In-Kind (PIK) program, the Agricultural Services Department of the Marine First National Bank in Janesville, has advised its customers to stay out of the market until it levels off, according to James Atchinson.

Iowa farmer, John Holst, indicates that weekly charts give him a true indication of what the pricing trend will be and confirm to him what the daily charts have been telling him.

SPOT TRENDS: MOVING AVERAGES CHARTS:

The moving average chart plots two or more averaged figures over a given period of time. These charts plot 4-day, 9-day, and 18-day moving averages.

With the information obtained from these charts, a determination can be made as to whether buying pressure is consistently stronger than selling pressure, or vice versa—selling pressure is stronger than buying pressure, and, thus, which way the major trends are pointed.

OVERBOUGHT/OVERSOLD: RELATIVE STRENGTH INDEX CHARTS:

The Relative Strength Index (RSI) chart is used to indicate when markets become overbought or oversold. This index will often accurately indicate when markets are turning either up or down, or giving an entry or exit signal. But, it will not provide any indication of when to reverse any previously taken position. It is, therefore, best used as a tool in combination with other technical trading systems and with fundamental information.

MEASURE OF TRADING ACTIVITY: VOLUME CHARTS:

This barometer of trading activity shows the number of trades in all futures contracts for any particular commodity on a given day. As a barometer of trading activity and an indication of the type of buying, volume charts, combined with open interest and bar charts, often show changes in market conditions which are not necessarily connected with apparent developments in supply and demand of the cash commodity. Thus, volume is important to watch, because changes in volume are often associated with price changes in the same direction. For example, a gradual increase in volume during an uptrend suggests a further price rise, whereas, a rapidly accelerating volume following a substantial upward price movement often signals a major top and a possible price reversal.

Charts in Agri-Star for volume of trading are drawn per commodity, not per contract, as most other charts are.

MARKET CHANGES: OPEN INTEREST:

As a barometer of trading activity, and an indication of the type of buying, open interest often can show changes in market conditions which are not necessarily connected with apparent developments in supply and demand of the cash commodity. For example, new export business may be reflected in futures markets before it is publicly announced, because it results in increased open interest. This is caused by commercial exporters buying futures contracts of the commodity they are going to export to protect themselves against price increases.

Open Interest charts in Agri-Star are drawn per commodity, not per contract, as most other charts are.

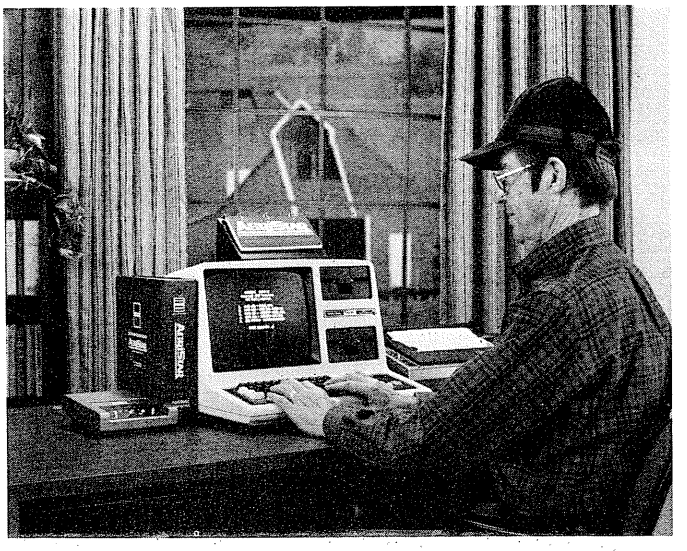
SPOT SIGNALS: POINT & FIGURE CHARTS:

(To be added in the near future). A point and figure chart gives a clear indication of when the market is making either higher highs or lower lows. Thus, the confirmation of a downtrend, for example, would provide a signal to the hedger to either sell the cash commodity or to get short in the futures market, or both.

Unlike bar charts, however, the point and figure charts signal specific price points for buying and selling the commodity being charted.

KNOWING WHAT CHARTS ARE AVAILABLE ON AGRISCAN

Help screens for Top Farmer Charts (TFC) and Top Farmer Historical Charts (TFH) are available under the AgriScan Help Menu, with the general heading: "Using Technical Charting Tools..HELP12." A "submenu," specific to the different types of Agri-Star charts, will appear as part of this help screen.



(Continued on page 30)

Communications Corner

by Al and Dru Simon

Hello and welcome back to Communications Corner. Last month we discussed the details and functions of each part of the RS-232-C connector, hopefully, learning "Everything we wanted to know about RS-232-C but were afraid to ask." As a follow-up to what we have discussed, this month's column might be appropriately entitled, "Well, Now That I Know All That, What Good Does It Do Me?" In response to that question—"A Lot!" Let's talk about it.

Once you have learned what the functions of the pins are (as described last month) and you have learned how to control them (as we'll describe this month), you will be able to interface just about any piece of equipment to any other piece of equipment. As an example, one of my customers had a problem because they wanted the information contained on their micros put onto their mainframe. This was no easy task, because they had Apple IIs, Apple IIIs, Wang Word Processors, a Burroughs mainframe, and Radio Shack Model 1s and Model 2s. The Apple representative said "It can't be done." The Burroughs man said "Can't do it!" I said "Why not?" I stepped in, and, as a result, my customer now enjoys full communications between all his equipment! You only have 25 pins! Once you know how to control those 25 pins there is NO serial device that you can't interface to any other serial device!

The object of the column this month is to teach you how to make up special cables to connect two odd pieces of equipment to each other where no off-the-shelf cable is available. You will learn a little bit about cable theory, and you will definitely learn how to connect your computer to that serial peripheral that you just bought. It is all very simple because the theory behind it is as obvious and clear as a sunny day!

As an example of some of the peripherals that you might be interested in interfacing, let me rattle off a short list: Modems, Scanners, Printers, Terminals, and so forth. The RS-232-C can also be used as a simple device control.

One of the first things you should know about RS-232-C is the fact that it's a PORT, and that differs from a memory address in the following manner: A memory address is a location which is contained in RAM (Random Access Memory) and can be addressed or read, and data can be stored in it. A port is simply a hard-wired device for passing information in or out. It is not addressable in the same manner that memory is in that you cannot store anything in a port location. You can send data out a port or read a port, but you cannot store anything in a port.

TO CONNECT TWO COMPUTERS TOGETHER

Let's get down to the business at hand. Now that we know what all the parts of our connector do, what interesting and useful things can we do with it? Let's start with the basics—the easiest of all applications for an RS-232-C would be as an interconnecting cable to connect two computers together. I am often asked how this can be done. Let's say that you have a Model 16 and a Model 3, and you want to move some programs from one to the other without going over telephone lines. It does seem rather strange to have to phone the computer in your living room from the computer in your bedroom! Sounds more like something your teenager might do—and you need to get those games onto the other computer so your teenager will leave your computer alone! The easiest way to accomplish the transfer of programs would be for you to make up a cable which will allow you to connect the two machines together so that you will need no modem at all. This type of cable is cryptically called a Modem Eliminator.(Oh, this computer terminology can be complex and mysterious sometimes, can't it?)

The job of creating this cable can be done very easily. If you refer to last month's chart, you'll see that pins 2 and 3 are Transmit and Receive Data. Pins 4 and 5 are Request to Send Data and Clear to Send Data. Those two pairs of pins always go together, sort of like salt and pepper shakers do. There are also 3 more pins which you have to consider, namely the Data Set Ready, pin (6); Carrier Detect, pin (8); and Data Terminal Ready, pin (20). These three pins also have to be present to accommodate most software. Remember that this is a software controlled item, not a hardware controlled item. If your software is looking for a carrier detect, then you have to have one. Most programs which run on Radio Shack computers require these three signals.

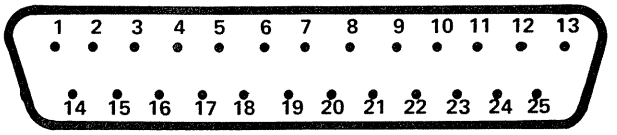
The easiest way to insure that you are getting these three signals when you are making a Modem Eliminator (or Jumper Cable as it is also called—make you think of your car's battery?) is to tie pins 6, 8, and 20 together at both ends of the DB-25 connectors. Most often you will be using male connectors at both sides of your ribbon cable, and the easiest way to jumper these together is by opening your connector, taking a piece of 30 gauge KYNAR (which is a type of very thin insulated wire which is readily available and usually in its own special container that cuts and strips the wire for you) about two inches long, stripping it, but leaving about 1/4 to 3/8 of an inch of insulation on the wire. Starting from the inside of the connector working toward the outside, place the bare KYNAR on pin 6, slide down the piece of insulation against pin 6 so that it protects pin 7, and put the piece of KYNAR from the outside of pin 8 to the inside on pin 8. From pin 8 going straight across, you will run into pin 20. Bring the KYNAR through to pin 20, and clip off the excess. Now when you lay your ribbon cable over it, it will secure the KYNAR into place when the connector is closed.

Please remember that in order for this cable to allow your computers to "talk" to each other the Receive must connect with the Transmit!! Now since you must not connect Receive to Receive or the Transmit to Transmit you must remember to reverse wires 2 and 3 on the other side of the cable! If you

look at the connector and look at the ribbon cable you're working with, you will notice that the cable is laid out as follows:

1 14 2 15 3 16 4 17 5 18 6 19 7 20 8 21 9 22 10 23 11 24 12 25 13

In fact, if you look at the DB25 itself, you will see the numbers of the wires in the ribbon cable listed on the connector, set up as follows:



Notice that you start off with 1, 14, 2, etc. so when you're looking for a particular pin location on the ribbon cable you must count every other wire, because every other wire goes to the bottom half rather than the top half of the connector. When locating wires, you must be exceedingly careful that you have located the correct one! Now, we wish to reverse wires 2 and 3 in order to make our Transmit wire go into our Receive wire and vice versa, so count carefully and locate wires 2, 3, and 15. (15 will be in the middle, going to the lower half of the connector). Once you are certain you have located the correct three wires, split the cable there carefully; cutting it just a bit with a razor to separate these three wires from all the rest. Take these three wires (which will be 2, 15, and 3) and just twist them around so that the #2 wire is where the #3 wire was; the #15 wire remains unchanged in the middle, and the #3 wire is where the #2 wire was.

Count carefully again and locate wires 4, 17, and 5. Reverse these wires as you did above with wires 2, 15, and 3. Press the wires onto the pins. The balance of the wires can be pressed onto the pins in their natural order. Go to the other side of the connector and tie together pins 6, 8, and 20 as you did on the first end of your cable, then just lay the ribbon cable in on top and put the connector over it. Close the connector and trim it. You now should have a connector with two sections where wires are reversed.

SOME HINTS FOR CABLE MAKERS

The easiest way to RUIN a cable is to put it on and take it off a connector repeatedly, so you must try to do the job carefully and accurately the first time.

There is a very easy way to seat a wire in a pin, and we recommend that you use this method. You will need a "pin setter" to seat the wire into the pin without damaging either. The "magic" tool that we use as a "pin setter" is the front of a ball point pen! That's all it takes, folks! Just lay your cable across the pins and while holding it tightly around the connector, rub the PLASTIC barrel of the pen (use plastic so you don't damage the pins) across the cable until you BEGIN to see the point of the pin sticking through the insulation. At this point you take the front of the pen, and one by one put each pin inside the barrel of the pen. This will push the cable down around the pin without damaging the pin itself. Care should be taken to lay the cable onto the pins very carefully otherwise you will be causing shorts between the pins.

Once your connector is completed you can plug one end into each of your computers and transmit programs at 9600 baud or whatever speed you like best. Your two computers are now connected to each other!

CONNECTING A COMPUTER TO A MODEM

To connect a computer to a modem, all you have to do to make a cable is simply lay in the cable as is. When you make up this cable, you must be certain that you identify pin 1 on your cable with a stripe or other marking to make sure that it goes to pin 1 on both connectors.

Please note: The above is true for all of the Radio Shack computers except the Color Computer, where you MUST tie in the pins 6, 8, and 20 if you are using a NON-Smart modem. The Modem II and DC 1200 are Smart Modems—all other modems must have the 6, 8, 20 jumper installed at the modem end of the cable. Unless—as in the Modem 1—there is a force DTR switch.)

WHAT IF YOU HAVE A NON-RADIO SHACK PRODUCT?

"But, Al," you say. "I have a product someone gave me. How do I get it to interface with my Radio Shack equipment?" My answer is simple. READ THE MANUAL THAT CAME WITH THE EQUIPMENT!! Within the manual you will find "pin outs" for the serial port (in other words—the description of each pin's function). They may be set up differently from your Radio Shack equipment, but there will never be more than 25 pins, each with its own signal. You already know (from last month's article) the function of each signal. All you have to do is match signals from one side of the cable to the other side.

Let's say you have an unusual cable that connects to your printer where perhaps Receive Data is pin 1 instead of the usual pin 3. All you have to do is make up a cable that will match the proper pins to the proper functions. Having done this you're ready!

CHECKING OUT YOUR CABLE

I do advise though, that after you have made a cable you take a volt/ohm meter and make certain that your cable does not have any shorts in it. If you don't have such a meter a simple 1-1/2 volt battery and light bulb will do. Wire the bulb to the battery on one terminal; take the other terminal from the battery and the other terminal from the light bulb and extend wires from them. Touching these 2 wires to anything which is continuous will cause the light to light, so if you want to check any connections that you have made, simply touch your two wires to it. You ought to be able to make the light light up. If it doesn't then reset your pins; push down a little further on the plastic because the connection was probably not made correctly.

HOW DO YOU CONTROL THE PINS?

Armed with above new skills let's talk about how you control each of these pins. There are three methods with Radio Shack equipment of controlling these pins. The first method is through machine language Service Calls (or ROM calls). Please refer to the technical sections of your Manual to locate and find syntax for these calls. There you will find machine language service calls for the Models 2, 12, and 16, and machine language ROM calls for the Color computer, and Models 1 and 3. The manual will define what they are and how to use them.

In the case of direct port addressing for the Models 1 and 3 please refer to Figure 1. In the case of the Color Computer, since there is no UART (a device which turns parallel data into serial data), all output must be manually configured and

timed and sent directly to the port. Please refer to your Color Computer technical manual for port addresses and methods.

When dealing with Models 2, 12, and 16 for port addresses, please see Figures 3 and 4.

Figure 1.

DATA BIT	MODEM STATUS REGISTER	CONFIGURATION SENSE SWITCHES	UART CONTROL REGISTER AND HANDSHAKE LATCH	UART STATUS REGISTER
D7	Clear to send Pin 5 DB-25	Even Parity Enable 1 = even, 0 = odd	Even Parity Enable 1 = even, 0 = odd	Data Received 1 = Condition true
D6	Data Set Ready Pin 6 DB-25	Word length Select 1	Word length Select 1	Transmitter Holding Register Empty 1 = Condition true
D5	Carrier Detect Pin 8 DB-25	Word length Select 2	Word length Select 2	Overrun Error 1 = Condition true
D4	Ring Indicator Pin 22 DB-25	Stop Bit Select 1 = 2 bits, 0 = 1 bit	Stop Bit Select 1 = 2 bits, 0 = 1 bit	Framing Error 1 = Condition true
D3	Unused	Parity Inhibit 1 disables parity	Parity Inhibit 1 disables parity	Parity Error 1 = Condition true
D2	Unused	Baud Rate 3	Break 0 Disables Transmit data	Unused
D1	Receiver Input UART P20	Baud Rate 1	Data Terminal Ready Pin 20 DB-25	Unused
D0	Unused	Baud Rate 2	Request to send Pin 4 DB-25	Unused
	IN ØE8H	IN ØE9H	OUT ØEAH	IN ØEAH

BIT ALLOCATIONS FOR REGISTERS AND LATCHES

The information in Figure 1 can be found on page 17 of the manual for the Model 1 RS-232-C board (26-1145).

Figure 2.

NIBBLES FOR BAUD RATE CONFIGURATION
ON FIGURE 1

Nibble Loaded		Transmit or Receive Baud Rate
HEX	BINARY	
0H	000	50
1H	001	75
2H	010	110
3H	011	134.5
4H	100	150
5H	101	300
6H	110	600
7H	111	1200

The information in Figure 2 is located on page 16 of the Model I RS-232-C interface board manual (26-1145).

For Models 2, 12, and 16 there are three read registers for Channel B and two read registers for Channel A within the SIO which can be used to obtain status information. Read registers 0 through 2 (Figure 4) contain the interrupt vector which is only valid for the Channel B register. In order to read the contents of a particular read register other than read register 0, your system program must first write the register number which you choose to look at to write register 0. Then by doing an IN to that register, the contents of that register will be returned.

Within the SIO there are seven write registers for channel A and eight for channel B which are programmed separately to configure the function of the channels. The interrupt vector for both channels is contained in write register 2 and is only in the channel B register set. The write registers require two bytes (excepting write register 0). The first byte is to write register 0 and contains three bits that point to the selected register. The second byte is the actual control word written into the register to configure the SIO. Following are two example usages. The first is in BASIC, and the second is in Machine Language.

Figure 3

PORT ADDRESS LOCATION
FOR MODELS 2, 12, AND 16

Port No.	Allocation	Function
F4H	SIO A	Channel A Data
F5H	SIO B	Channel B Data
F6H	SIO A	Channel A Command/Status
F7H	SIO B	Channel B Command/Status

The information in Figure 3 is found on page 20 of the Model II Technical Reference Manual (26-4921). It is applicable for the Models 12 and 16 as well.

The BASIC example:

To set the DTR in the Models 1 and 3, do an OUT 234,1. Use the table in order to see which bits must be set.

For example, in order to set the number of stop bits to "1" with the larger machines, you would do the following:

```
OUT  &HF6,4   (attention register 4!!!!!)
OUT  &HF6,4   (set one stop bit)
```

The machine language (Z80) example:

```
LD   B,0F6H  'Load control register address into B
OUT  (B),04H  'Attention register 4 !!
OUT  (B),04H  'Set bit 2 (one stop bit)
```

To further illustrate:

Figure 4. Read and Write Bit Function Tables

READ REGISTER 0

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
								Rx CHARACTER AVAILABLE
								INT PENDING (CH. A ONLY)
								Tx BUFFER EMPTY
								DCD
								SYNC/HUNT
								CTS
								Tx UNDERRUN/EOM
								BREAK/ABORT

*Used With External Status Interrupt Mode

READ REGISTER 1†

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	I FIELD BITS IN PREVIOUS BYTE	I FIELD BITS IN SECOND PREVIOUS BYTE
									ALL SENT
			1	0	0			0	3
			0	1	0			0	4
			1	1	0			0	5
			0	0	1			0	6
			1	0	1			0	7
			0	1	1			0	8
			1	1	1			1	8
			0	0	0			2	8
									PARITY ERROR
									Rx OVERRUN ERROR
									CRC/FRAMING ERROR
									END OF FRAME (SDLC)

*Residue Data For Eight Rx Bits/Character Programmed

†Used With Special Receive Condition Mode

READ REGISTER 2

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
								V0
								V1†
								V2†
								V3†
								V4
								V5
								V6
								V7

INTERRUPT VECTOR

†Variable If Status Affects Vector Is Programmed

Read Register Bit Functions

WRITE REGISTER 0

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
				0	0	0		REGISTER 0
				0	0	1		REGISTER 1
				0	1	0		REGISTER 2
				0	1	1		REGISTER 3
				1	0	0		REGISTER 4
				1	0	1		REGISTER 5
				1	1	0		REGISTER 6
				1	1	1		REGISTER 7
			0	0	0			NULL CODE
			0	0	1			SEND ABORT (SDLC)
			0	1	0			RESET EXT/STATUS INTERRUPTS
			0	1	1			CHANNEL RESET
			1	0	0			ENABLE INT ON NEXT Rx CHARACTER
			1	0	1			RESET TxINT PENDING
			1	1	0			ERROR RESET
			1	1	1			RETURN FROM INT (CH-A ONLY)
						0	0	NULL CODE
						0	1	RESET Rx CRC CHECKER
						1	0	RESET Tx CRC GENERATOR
						1	1	RESET Tx UNDERRUN/EOM LATCH

WRITE REGISTER 4

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
								PARITY ENABLE
								PARITY EVEN/ODD
					0	0		SYNC MODES ENABLE
					0	1		1 STOP BIT/CHARACTER
					1	0		1½ STOP BITS/CHARACTER
					1	1		2 STOP BITS/CHARACTER
				0	0			8 BIT SYNC CHARACTER
				0	1			16 BIT SYNC CHARACTER
				1	0			SDLC MODE (01111110 FLAG)
				1	1			EXTERNAL SYNC MODE
						0	0	X1 CLOCK MODE
						0	1	X16 CLOCK MODE
						1	0	X32 CLOCK MODE
						1	1	X64 CLOCK MODE

WRITE REGISTER 1

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
								EXT INT ENABLE
								Tx INT ENABLE
								STATUS AFFECTS VECTOR (CH. B ONLY)
				0	0			Rx INT DISABLE
				0	1			Rx INT ON FIRST CHARACTER
				1	0			INT ON ALL Rx CHARACTERS (PARITY AFFECTS VECTOR)
				1	1			INT ON ALL Rx CHARACTERS (PARITY DOES NOT AFFECT VECTOR)
								WAIT/READY ON R/T
								WAIT/READY FUNCTION
								WAIT/READY ENABLE

*On Special Condition

WRITE REGISTER 5

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
								Tx CRC ENABLE
								RTS
								SDLC/CRC-16
								Tx ENABLE
								SEND BREAK
						0	0	Tx 5 BITS (OR LESS)/CHARACTER
						0	1	Tx 7 BITS/CHARACTER
						1	0	Tx 6 BITS/CHARACTER
						1	1	Tx 8 BITS/CHARACTER
								DTR

WRITE REGISTER 2 (CHANNEL B ONLY)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
								V0
								V1
								V2
								V3
								V4
								V5
								V6
								V7

INTERRUPT VECTOR

WRITE REGISTER 6

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
								SYNC BIT 0
								SYNC BIT 1
								SYNC BIT 2
								SYNC BIT 3
								SYNC BIT 4
								SYNC BIT 5
								SYNC BIT 6
								SYNC BIT 7

*Also SDLC Address Field

WRITE REGISTER 3

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
								Rx ENABLE
								SYNC CHARACTER LOAD INHIBIT
								ADDRESS SEARCH MODE (SDLC)
								Rx CRC ENABLE
								ENTER HUNT PHASE
								AUTO ENABLES
						0	0	Rx 5 BITS/CHARACTER
						0	1	Rx 7 BITS/CHARACTER
						1	0	Rx 6 BITS/CHARACTER
						1	1	Rx 8 BITS/CHARACTER

WRITE REGISTER 7

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
								SYNC BIT 8
								SYNC BIT 9
								SYNC BIT 10
								SYNC BIT 11
								SYNC BIT 12
								SYNC BIT 13
								SYNC BIT 14
								SYNC BIT 15

*For SDLC It Must Be Programmed to 01111110 For Flag Recognition

Write Register Bit Functions

```
LD    B,0F6H  'Load control register
              address into B
OUT   (B),04H 'Attention register 4 !!
OUT   (B),08H 'Set bit 3 (2 stop bits)
```

In the case of the Models 1 and 3 the first OUT command is not necessary and B register should first be loaded with port address you desire.

Using the above tables you can see how you can control each of the pins in the DB25 connector. If you have specific questions, please contact your local service representative or write to us in care of Communications Corner, P.O. Box 2910, Fort Worth, Tx 76113-2910. You now should have enough new knowledge and skills to be able to interface and program any RS-232-C peripheral device.

PARTS ARE AVAILABLE

It is possible to purchase the parts needed to build your cables from Radio Shack.

Connectors

D-Submini 25 Male for RS-232C bus
(276-1559) Sug. Ret. $4.99

D-Submini 25 Female - Ideal for Cable Extensions
(276-1565) Sug. Ret. $4.99

Solder D-Submini 25 Connectors

Male	(276-1547)	Sug. Ret. $2.99
Female	(276-1548)	Sug. Ret. $3.99
Hood	(276-1549)	Sug. Ret. $2.19

Cable and Wire

Flat RS-232C 25 Conductor 5′ Cable (278-772)
Should be available June/July Sug. Ret. $3.59
KYNAR, 30 gauge wire Sug. Ret. $2.39
Red (278-501)
White (278-502)

AND IF YOU DON'T WANT TO BUILD THE CABLE . . .

For those who don't want to build their RS-232-C accessories, Radio Shack offers the following RS-232-C cables and accessories.

	Cat. No.	Sug. Ret.
Color Computer to RS-232C Interface	(26-3014)	$19.95
Model III/4/12 RS-232C 5′ Flat Cable	(26-1408)	$19.95
Model III/4 8″ Extender - Allows use of round RS-232C cables with the Model III/4	(26-1497)	$17.95
Model II,16 RS-232C 5′ Cable with Terminator (Round with molded plugs)	(26-4403)	$39.95
RS-232C Cable for Model 12 or for extending any RS-232 Cable		
10′	(26-1490)	$29.95
25′	(26-1491)	$39.95
50′	(26-1492)	$54.95
100′	(26-1493)	$89.95
RS-232C Cable Ext. for Longer Runs	(26-1495)	$29.95
Null Modem Adapter—Allows two CPUs to communicate directly using modem software	(26-1496)	$29.95

THE CORNER MAILBOX

I have always liked the idea of trading user made programs, but all the phone numbers are not in my area. If you know of any in my area or know how to get a catalog of numbers, please send it to me. Also do I need an RJ11C modular jack for a Modem II?

Buc Burgess
Altamonte Springs, Fl

Dear Buc,

Although our January article did not specifically mention bulletin boards in your area, we know that there are plenty in Florida. Call any of the boards listed in the January or April articles, and refer to their lists of other bulletin board numbers. Keep a hard copy of these lists and refer to them as need arises. You will probably find new boards listed on each one that you call.

About your Modem II—yes, you do need the RJ11C modular jack.

Although I can effectively use my Model III as a terminal, there does not seem to be much software available for running the modem. The IBM system uses an 80 character CRT while the Model III is only 64. I would like to eliminate the wrap around problem. The first idea I had was to route the information from the RS-232-C to my line printer instead of the screen. The 1.3 version of TRSDOS, however, returns ERROR 31 (program not found) whenever I attempt to use the ROUTE command.

Steven C. Rowland
Monterey, Ca

Dear Steven,

You idea is a good one. However, it is quite true that the ROUTE command has been replaced! The best way to accomplish what you're trying to do is to open your buffer, capture the incoming information as a disk file and then print the disk file. However, if you should want to use your printer, you must POKE &H4222 with a two byte ASCII abbreviation of the desired source device (one of KI,DO,RI,RO,or PR) and &H4220 with a two byte destination abbreviation. See page 50 of your manual to see how to do this from BASIC.

I am using a Model III with 2 disk drives to communicate to an HP3000. I notice communications package does not seem to be using the $RSTX and $RSRCB ROM calls. They seem instead to be talking to ports E8 through EB instead. Your assistance is appreciated.

Dr. V. Zitko
St Andrews, New Brunswick

Dear Dr. Zitko,

You are quite correct in that the terminal package does not use the ROM calls. Instead it directly addresses the ports as the above article describes. If you have any further questions about it, please write to us again.

That about wraps it up for another month in our corner, please watch for our next month's article in which we will discuss Direct Cursor Addressing, Communicating Graphics, and methods of printer dumping. Happy Communicating!

Color PILOT

by George Gerhold

Editor's note: Color PILOT was developed for the TRS-80 Color Computer by Dr. George Gerhold and Mr. Larry Kheriaty of Western Washington State University. Kheriaty and Gerhold were the original developers of COMMON PILOT, a widely used version of the PILOT language.

Color PILOT is a language used to create instructional dialogs on the TRS-80 Color Computer. Because Color PILOT makes it easy to code flexible screen design and answer processing, the language can be used by teachers and curriculum developers who are not computer experts. Color PILOT is a version of COMMON PILOT, which means that instructional programs written in PILOT for many other machines can be converted for use on the Color Computer very easily.

Why use Color PILOT instead of BASIC for instructional programming? That's like asking a fence builder, "Why use a post-hole digger when shovels are available?" In both cases, the answer is, "because your productivity is higher and the product is better when you use a tool designed for the task at hand." The larger the project, the more important this becomes.

Instructional programming and computational programming are very different tasks. Good instruction involves transmitting a large quantity of textual and graphic information in very digestible form. It also requires the ability to accept and correctly process a wide variety of student input. Color PILOT makes it easy to code (program) common instructional operations. For example, students are likely to make spelling and typing errors, and they are likely to use a variety of words and expressions in responding to questions. Color PILOT makes it easy for you to allow for such variety in responses; BASIC does not. On the other hand, BASIC makes it easy to code loops, which repeat operations many times. In instruction, however, we seldom wish to repeat the same thing many times, so Color PILOT does not have such an operator.

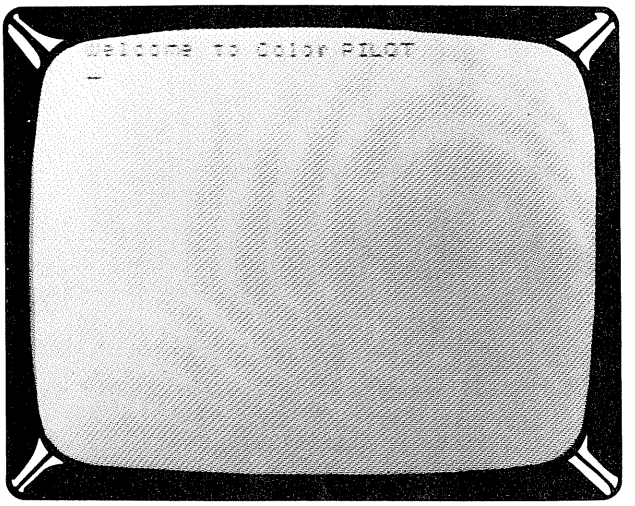
Let's take a look at how Color PILOT helps the instructional programmer. We'll begin with output. Think of the computer's display as a blackboard, not as a piece of paper. We do not have to fill the screen with information. We can write on it anywhere we please. We can use the whole screen for a single idea, or we can divide it into sections; we can mix graphics and text; we can write in large and small letters; we can make any characters we need; we can erase selected portions while keeping others; and we can use an unlimited number of panels at little extra expense. All of these things can be done using Color PILOT.

In Color PILOT, you use two main instructions to generate displays: the TYPE instruction and the GRAPHIC instruction. The following TYPE instruction (with the computer understanding uppercase "T" as the code for the instruction), will display the message "Welcome to Color PILOT" on the screen:

```
T:Welcome to Color PILOT.
```

Everything after the colon in the TYPE instruction appears on the screen, at the current cursor position, as typed. (Notice that you can use upper and lower case in Color PILOT.) Modifiers can be added to the TYPE instruction to affect the way messages are displayed on the screen. The "S" modifier (for "Screen") clears the screen, then prints the message starting at the screen's upper left corner:

```
TS:Welcome to Color PILOT.
```



Many of the display enhancements for text are controlled via the GRAPHIC instruction ("G"). You can use this instruction to change color sets, to select normal or reverse video, and/or to use double-size characters. By combining these options in different ways, you can get a total of eight different screen modes. The instruction:

```
G:M6
```

selects mode 6—namely, the alternate color set, normal video, and large characters.

By using modifying operators with the GRAPHIC instruction, you can specify the screen coordinates for text display or for drawing graphics. The screen is treated as a piece of graph paper, and positions are specified by an X,Y pair of numbers or variables. The origin is in the lower left corner, unless moved. To position text on the screen, we use the WINDOW ("W") operator. The following instruction will position the text cursor in the center of the screen:

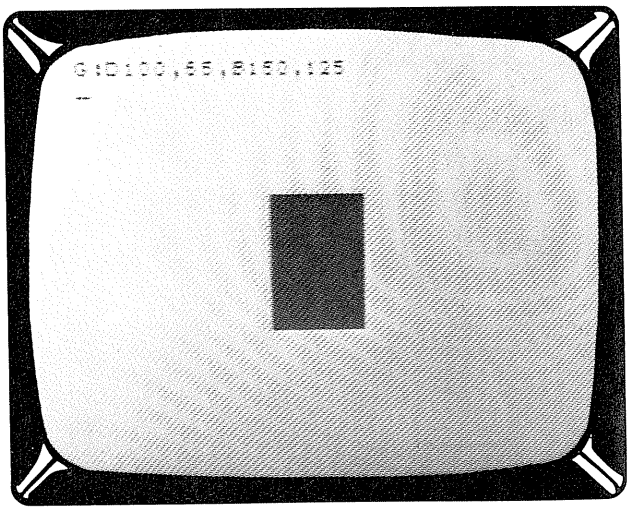
```
G:W126,96
```

It also reduces the size of the text screen by making the indicated point the upper left corner of the text window. All of the TYPE instructions to follow (including TS: instructions) will operate only within the window. This allows the author to lock part of the display on the screen while changing other parts. Of course the window can be reopened with another WINDOW operation.

The BLOCK operation ("B") presents another way to selectively erase the screen. To use the BLOCK operation, we have to designate two points on the screen—two opposite corners of a block. The instruction:

G:D100,65,B150,125

will draw a block approximately in the center of the screen.



The block can be drawn in any of four colors in either of two color sets. By making the block the same color as the screen's background, we have a way to selectively erase portions of the screen. Of course blocks can be used as components of graphics, in combination with the DOT, LINE, PENCOLOR, and ERASE operations that are available.

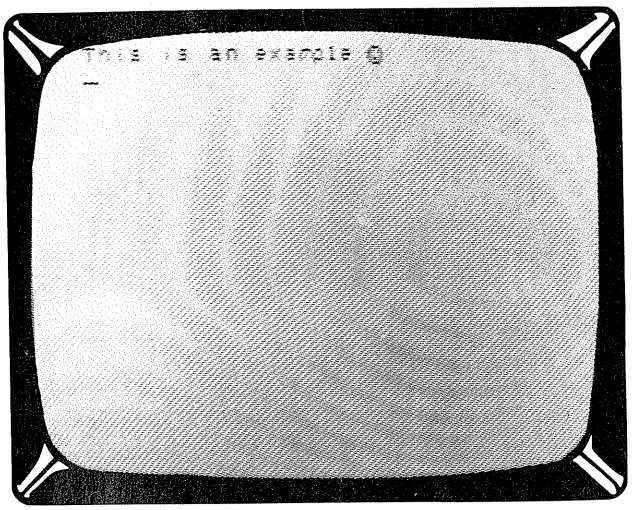
Another very useful feature of Color PILOT is the NEWCHARACTER instruction ("N"). This instruction allows you to custom-design characters for special instructional purposes and/or for use in graphic designs. You can redefine ASCII characters 32 through 127, giving them any 8×8 dot pattern. For example, the instruction:

N:32,FFFFFFFFFFFFFFFF

changes the space (character 32 in the ASCII convention) into a solid black box. The dot pattern is specified by 16 Hex digits—in this case, we have turned all of the dots ON. After this instruction is encountered in a program, every space will appear as a black box. Patterns can also be defined for characters that are not provided by the keyboard. For example, the instruction:

T:This is an example #127

will produce character #127 after the other text even though character 127 does not correspond to a key.



Now let's look at Color PILOT's capabilities for processing student input. When the student responds to a question, his or her input is immediately transferred into a buffer. Various MATCH instructions can then be used to make the computer search the buffer for input that matches the specified range of acceptable answers. For example, the instruction:

M:senate

causes the computer to search the buffer for the answer "senate." ("M:senate" causes a window-string search, meaning that the answer "senate" will be found even if it appears in a sentence or in a group of words.) Color PILOT provides many extensions to the power of the MATCH operation. The computer can be told to convert all student input either to upper case or to lower case before comparison (in the buffer only, not on the screen). The items to be matched can contain single wild-card characters (*); multiple wild-card characters (& can be used to join two required parts of an answer, with anything accepted in between these parts); and OR operators (!). The instruction:

M:sen*t!h&se!leg*sl&t*r!repr*s&t&v

would treat the answers "senate", "house", "legislature", and "representative" in a great variety of misspellings and mistypings as equivalent. The modified MATCH instruction:

MS:senate

would match any student answer which was only one character different from "senate" or which differed only by the switch of two characters.

Of course we must be able to take different actions depending on the student answer. By using Y or N conditions, we can make the execution of any instruction depend on the results of a MATCH operation. The following is a very simple example:

```
TS:Name the smaller of the
:national legislative bodies.
A:
M:senate!Senate
TY:That's right!
TN:No, that's not right.
```

(Continued on page 28)

Telidon: Graphics of the Near Future

Editor's Note: The CompuServe Information Service is one of the largest information and entertainment services available to owners of personal computers and computer terminals. With each issue of TRS-80 Microcomputer News, various features of CompuServe will be discussed. The CompuServe Information Service is sold at Radio Shack stores nationwide and in Canada.

An architect meeting with her clients uses it to take them on a visual tour of a building that is still on the drawing boards.

A chemist uses it to create models for investigating the interaction of large molecules.

A stockbroker gets an almost instantaneous graph of the performance of various stocks on which to base his financial judgments.

These and many other examples in the fields of entertainment, medicine, business and education are all current uses of computer graphics—a science that has moved in the past few years from one of novelty to serious research.

According to Larry Sturtz, manager of CompuServe's Advanced Research Department, new developments in computer graphics will continue to occur rapidly. "All this work will eventually benefit home computer owners," he says. "Initially, the software will be made available so that personal computer owners can use BASIC or FORTRAN programs to generate graphics. Later, computer products will be released featuring Telidon graphics and requiring no technical knowledge."

In the forefront of computer graphics development is the Canadian Telidon Videotex system, a new terminal-independent method of creating high-resolution, multi-color graphic images through the use of Picture Description Instructions (PDIs). This PDI code describes images in terms of commands to draw basic graphic shapes such as lines, arcs and polygons at specified positions in the overall picture.

Sturtz, who is heading CompuServe's development of software to support the Presentation Level Protocol (PLP) which is a subset of the Telidon system, says that his group's work is a continually-evolving process.

"I think we're close to the point where software will be available for use by "quasi-technical" people. We have one commercial customer in New York to whom we've supplied software for use with his own database," Sturtz explains.

"Telidon is not widely used at this point," he adds, "but we believe—after evaluating other systems—that it's the one that will become the North American videotex standard."

CompuServe's Advanced Research team displayed their work on Telidon graphics with a stock data demonstration at Interface '82, a trade show held last March in Dallas. An entirely computer-generated program, the software requires no artist or information provider to draw pictures—a real plus in eventually making it available to the business person or home computer hobbyist.

What the system does require at this point, however, is extremely skilled people to work with the Telidon software/hardware combination during the graphics creation process. According to Charles Csuri, professor of Art Education and Computer Science at the Ohio State University, there is a "crying need" for software that will make computer graphics a viable option for business and home use. "The hardware will continue to go down in price," Csuri says, who is nationally known for his pioneering efforts in computer art. "The real deterrent is software—packages must be generated to solve very specific problems generated by users' needs."

Sturtz agrees with Csuri's assessment of the state-of-the-art, adding that research on software for the Telidon graphics system and others is an ongoing priority at CompuServe.

"We're working now on one of several graphics projects, to develop a multi-user interactive game that can be used by home computer hobbyists," he says, adding that developments of this sort take months—even years—of research before they are ready for the public.

Some synthetic images produced by computer graphics cannot readily by distinguished from pictures of real scenes. In pilot training, computer graphics are already being used as a safe, efficient method to present students with ground-based experience that mimics precisely the view from a cockpit.

Use of computer graphics is also being experimented with as a means of eliminating the more tedious parts of cartoon animation. And instead of relying on artists to produce time-consuming and complicated charts, graphs and slides by hand, some businesses are relying on computer graphics to do the job.

Sturtz and other researchers, who spend long days in the lab working to make image generation procedures efficient, agree that working to simplify these processes is one of the greatest challenges in computer graphics.

OSU's Csuri predicts wide use of computer graphics as it moves from research labs into homes, offices and schools.

"Computer graphics will alter how we work, learn and play," Csuri says. "Its impact on all our futures will be enormous."

COLOR AND BLACK AND WHITE GRAPHICS AVAILABLE FOR STOCK DATA ON COMPUSERVE

Color and Black and White graphics for stock data is now available on the CompuServe Information Service for the TRS-80 Color Computer, TRS-80 Videotex and the TRS-80 Model I and III.

The microcomputer must be operating a CompuServe Vidtex-compatible executive program available through CompuServe.

At the MicroQuote program prompt, one can enter the VIDPLT command by answering the three questions of ticker, time period (daily, weekly or monthly) and ending date receive a graphic display of the high, low, close and volume on any stock.

Seventeen periods of data are presented with each graphic. Each period of data will represent daily, weekly or monthly data according to the user's request. The cost is 15 cents per graphic (17 periods per graphic) with no additional data charge.

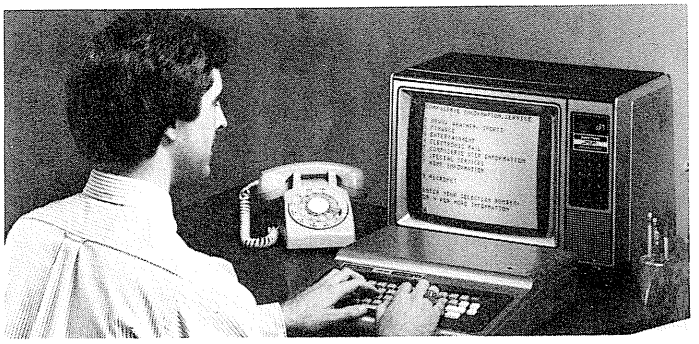
VIDPLT is an excellent means of viewing a stock's trend, especially for the home, amateur or small investor. CompuServe invites all users to view this graphics addition to the MicroQuote system which offers 10 years of daily information on over 42,000 financial instruments.

VIDPLT is an additional graphics capability to MicroQuote. Other similar capabilities are TELPLT which presents stock graphics for Telidon compatible microcomputers.

MicroQuote is available under the Reference Database section of the Business and Financial Services or by entering GO FIN-20.

CompuServe would like to express their thanks to *Today* magazine for permission to reprint the "Telidon: Graphics of the Near Future" story from the Sept./Oct. 1982 issue of *Today*.

Questions and comments about the CompuServe Information Service can be sent to Richard A. Baker, Editorial Director, or Jacqueline A. Farthing, Assistant Editor, CompuServe Information Service, 5000 Arlington Centre Boulevard, P. O. Box 20212, Columbus, Ohio 43220 or through Feedback, main menu item 5, CompuServe User Information.



Grid for the CGP-115

Otis D. Sullivan
Sullivan Studios Inc.
212 Fifth Avenue
New York, NY 10010

This is a program that I wrote for the CGP-115 Color Graphics printer. The program (or subroutine) creates a grid that can be used to create charts.

```
1 '*** GRID PROGRAM FOR THE CGP-115 COLOR GRAPHICS
    PRINTER ***
2 '*** BY OTIS SULLIVAN 212 FIFTH AVENUE NEW YORK
    10010 ***
3 '********************  SEPTEMBER 1982
    **********************"
5 LPRINT"BILLIONS"
    : LPRINT"OF DOLLARS"
10 LPRINT CHR$(18)
    : LPRINT"C0"
20 LPRINT"L0"
30 LPRINT"M25,0"
    : LPRINT"I"
35 FOR A = 0 TO 10
40 LPRINT"J399,0"
50 LPRINT"R-399,-24"
60 NEXT A
70 LPRINT"H"
80 LPRINT"J0,240"
90 LPRINT"R21,240"
100 LPRINT"R-300,10"
101 LPRINT"S0"
105 LPRINT CHR$(173)
110 LPRINT"100"
    : LPRINT
    : LPRINT" 90"
    : LPRINT
    : LPRINT" 80"
    : LPRINT
    : LPRINT" 70"
    : LPRINT
    : LPRINT" 60"
    : LPRINT
    : LPRINT" 50"
    : LPRINT
    : LPRINT" 40"
    : LPRINT
    : LPRINT" 30"
    : LPRINT
    : LPRINT" 20"
    : LPRINT
    : LPRINT" 10"
    : LPRINT
    : LPRINT" 0"
120 LPRINT"    '65    '67    '69    '71    '73    '75
    '77    '79    '81    '83
```

This program may be modified for the Color Computer by changing all LPRINTs to PRINT#-2,.

COLOR PILOT (From page 26)

The line which starts a colon is the continuation of the TYPE instruction. The "A:" is an ACCEPT instruction which gets the student input. The MATCH allows for capitalization because we did not use the case editing feature. The operations of the conditional TYPE instructions are obvious. The YES and NO conditions are just two of the many conditions that can be used. In fact any legal expression—either numeric, alphabetic, or mixed—can be attached to instructions as conditions. If the expression is true, the instruction will execute. If the expression is false, the instruction is skipped.

We cannot completely describe Color PILOT in this small space, so we have concentrated on those features which distinguish it from other languages. Color PILOT is a complete language with numeric and string variables, assignment instructions, branching instructions, and file access instructions. It is possible to write complex simulations as well as simple dialogs in the language.

The Color PILOT package is available in two forms: a disk version for 32K which includes all of the above features, and a cassette version which includes a subset of these features and which runs on a 16K machine. Both systems include a simple line-oriented editor which makes creation and revision of programs extremely simple. Finally, in comparison to other versions of PILOT, the system is extremely fast. Try it; you'll love it.

Notes on Previous Newsletters

JULY/AUGUST 1982

Fireworks

Gregg Ford
95 Eldred Ave.
Bedford, OH 44146

I would like to suggest the following additions to the program submitted by Franklin Pigounias. These will liven up the program a little.

```
31 SOUND 1,1
32 FOR X=1 to 2ØØ
     : NEXT X
49 SOUND 2ØØ,1
51 SOUND 21Ø,1
61 SOUND 22Ø,1
1Ø1 FOR X=1 TO 1ØØ
     : NEXT X
1Ø3 SOUND 1,2
121 SOUND 75,2
131 SOUND 13Ø,3
221 SOUND 1,1
     : SOUND 1,1
3Ø1 SOUND 12Ø,2
3Ø9 SOUND 75,2
311 SOUND 4Ø,2
319 SOUND 25,2
321 SOUND 33,2
499 SOUND 1,4
531 SOUND 75,3
541 SOUND 12Ø,3
551 SOUND 8Ø,3
561 SOUND 2ØØ,2
571 SOUND 1ØØ,2
581 SOUND 1ØØ,2
591 SOUND 16Ø,2
6Ø1 FOR X=1 TO 2ØØ
     : NEXT X
```

NOVEMBER 1982

Printing Titles

William B. Innes
724 Kilbourne Dr.
Upland, CA 91786

The program does not appear suitable for a LP VIII printer. The following program was worked out with TRS-80 Level 2.

```
1Ø INPUT "ENTER TITLE"; A$ 'EXAMPLE GIVEN ABOVE
2Ø L=LEN(A$)'NUMBER OF SPACES IN A$
3Ø T=4Ø-INT(.5*L)'SPACES ON LEFT OF TITLE TO CENTER
   FOR LP VIII PRINTER
5Ø LPRINT TAB(T)A$'PRINTS TITLE AT CENTER POSITION
6Ø FOR Y= T TO T+6*L 'THERE ARE 6 GRAPHICS PER SPACE
7Ø LPRINT TAB(Y)CHR$(18);CHR$(129);'PRINTS GRAPHIC AT
   Y
8Ø NEXT Y' LOOP TO COMPLETE GRAPHIC PRINT UNDER TITLE
85 LPRINT CHR$(3Ø);"" ' CONVERTS GRAPHIC TO DATA
   PROC. MODE
```

FEBRUARY 1983

Periods to Commas in Data Statements

Louis B. Kelley
Route 1 Box 20-A
Crescent City, FL 32012

Regarding the program by Edward M. Roberts, I would suggest, for faster execution, changing FOR Z = 17129 TO VARPTR(Z) to read FOR Z = 27129 TO VARPTR(Z). It is one of the most useful "tricks" I have used.

Thanks from both of us, my Model III and me.

Variable Swapping

Dennis Lee Bieber
1630 Ottawa Ct. Apt. C
Sunnyvale, CA 94087

The variable swapping routine by Steven Kaiser is extremely wasteful. Only one temporary variable is needed. The same result can be obtained by deleting statement 30 and changing statement 40 to read "40 A = B".

The routine which follows swaps two variables without using temporaries. (I am ignoring the dummy arguments in the DEF FN statement as it could be coded in-line.) FNX is an XOR (exclusive or) statement (a useful statement lacking in BASIC).

```
1Ø DEF FNX(A,B)=(A AND NOT B) OR (NOT A AND B)
2Ø INPUT "A,B";A,B
3Ø PRINT "THE NUMBERS TO BE EXCHANGED ARE ";A,B
4Ø A=FNX(A,B)
     : B=FNX(A,B)
     : A=FNX(A,B)
5Ø PRINT "THE EXCHANGED NUMBERS ARE ";A,B
6Ø END
```

MARCH 1983

Resistor Color Code Interpreter

Editor's Note: We sincerely apologize for the omission of the last ten lines of this program. The missing lines should read:

```
6ØØ IF D=11 THEN G=.Ø1
61Ø IF D=12 THEN G=.1
62Ø RETURN
63Ø END
64Ø AQ=""
     : INPUT"4TH BAND ";AQ
65Ø IF LEFT$(AQ,3)="SIL" THEN AB="1Ø% TOLERANCE"
     : RETURN
66Ø IF LEFT$(AQ,3)="GOL" THEN AB="5% TOLERANCE"
     : RETURN
67Ø LET AB="2Ø% TOLERANCE"
68Ø RETURN
69Ø END
```

AREAD–The PC-2 Automatic Read

by Peter Levy

One of the PC-2's more underrated (and less understood) functions is its automatic read—AREAD.

The fundamental use of AREAD is simple. It reads whatever is currently on the LCD display as data and assigns the resulting value to a specified variable. If the display contains "456.7" and AREAD X is executed, then a value of 456.7 is assigned to X.

There are a couple of restrictions on AREAD's use. First, the PC-2 does NOT evaluate expressions during AREAD as it does for INPUT. Whatever is read from the display must be useable to the PC-2 as-is. Hence, if a numeric variable is loaded through AREAD, the contents of the display must be numeric, such as 234.56. Expressions such as 3+4 or SQR(3) cannot be evaluated by AREAD and will result in the declaration of an error.

Second, if AREAD is used, it must be the very first statement of a line; that line must be labeled; and the line must have been reached by entry of a DEF-key.

The following program will demonstrate AREAD's application:

```
1Ø PRINT "ENTER NUMBER, PRESS DEF-A"
2Ø END
3Ø "A" AREAD X
4Ø PRINT "VALUE READ =";X
5Ø END
```

Note that the AREAD statement is the very first statement in its line (30); that line 30 is labelled ("A"); and that the program flow is designed so that line 30 is reached by its DEF-key label (DEF) (A). These conditions all must be met for AREAD to be used.

Not let's examine what happens when this program runs, and so learn exactly what goes on when AREAD is used.

First, ENTER NUMBER, PRESS DEF-A is displayed. After displaying this the PC-2 stops running the program and waits for the user to tell it that he's through reading the display, which he normally does by pressing (ENTER) or a DEF-key.

At this point, a numeric value is typed. Let's type in 32.174. As soon as the user starts pressing keys the display is automatically cleared, and then the PC-2 echoes the new keystrokes on the display as they are received. We wind up with 32.174 on the display, plus a cursor. The PC-2 is still waiting for permission to go ahead after its PRINT, since neither (ENTER) or a DEF-key has been pressed.

Now we press (DEF), then (A). The PC-2 jumps to the line labelled "A" and starts executing that line, which instructs it to AREAD X. The PC-2 does this by reading the characters on the display—which still holds the 32.174 previously typed—and assigning that value to the variable X.

Now the program tells us what was read:

```
VALUE READ = 32.174.
```

AREAD is generally used where a routine needs data on entry, and, for one reason or another, it isn't practical to transmit that value in a variable.

Suppose you have a program which can perform a large number of different operations—say a geometry application. You use DEF-key labels to get to the various routines easily. Now it turns out that each routine doesn't necessarily use the same variables—likely enough, especially if some are to use the results of others. It's pretty clumsy to use, say, INPUT X at one point, then start each and every routine with U=X or A=X or Y=X or whatever conversion is needed. The situation gets still worse if some functions want a string value and others a numeric one.

The solution is to use AREAD at the start of each routine: AREAD X, AREAD B$, or whatever. Now all you have to do is type the value and then press the needed DEF-key. Handy, isn't it?

AgriStar (From page 19)

SUMMARY

Charting is not a utopia in price forecasting. It is a strategic tool for any agribusinessman in his overall marketing program to be used in conjunction with other fundamental information available on Agri-Star. And, as with any tool, its usefulness depends on knowing how to use it and knowing its capabilities and limitations. With the aid of other information on charting, an agribusinessman can improve his decision-making performance. He will have the knowledge necessary to know when to buy and when to sell his commodities profitably.

Such information available includes:

1. Top Farmer Strategy (TFS) reports issued daily on Agri-Star. These reports give:
   a. A summary of the dominant factors affecting the market by individual commodity and by commodity groupings;
   b. A technical summary of the charts—their meaning and the future price outlook;
   c. An overall summary which gives a price outlook and why; and
   d. Recommendations of when to buy and sell and what to look for in the near future.
2. Access to Agri-Star HELP screens; and
3. "Charting Farm Markets," a Top Farmers of American Publication.

For more information about Agri-Star, a business information service of AgriData Resources, Inc., call toll free 1-800-558-9044 (Wisconsin residents call 1-800-242-6001) or write Warren Clark, Marketing Manager, AgriData Resources, Inc., 205 W. Highland Ave., Milwaukee, WI 53203.

Greek Flag

H.E. Boulware
4516 St. Andrew Lane
Bridgeton, MO 63044

This program for the PC-2 with printer/plotter attached draws a Greek flag.

```
5 "GREECE"CLEAR
10 GRAPH
     : GLCURSOR (10,0)
     : COLOR 1
     : LINE (10,0)-(200,0)
20 LINE (200,-0)-(200,-135)-(10,-135)-(10,0)
     : GLCURSOR (40,0)
     : LINE (40,0)-(40,-30)
30 LINE (40,-30) - (10,-30) - (10,-45) - (40,-45) -
     (40,-75) - (55,-75) - (55,-45)
40 LINE (55,-45) - (85,-45) - (85,-30) - (55,-30) -
     (55,0)
     : GLCURSOR (10,-134)
50 LINE (10,-134) - (200,-134) - (200,-133) -
     (10,-133) - (10,-132) - (200,-132) - (200,-131)
60 LINE (200,-131) - (10,-131) - (10,-130) -
     (200,-130) - (200,-129) - (10,-129) - (10,-128)
70 LINE (10,-128)  - (200,-128) - (200,-127) -
     (10,-127) - (10,-126) - (200,-126) - (200,-125)
80 LINE (200,-125) - (10,-125) - (10,-124) -
     (200,-124) - (200,-123) - (10,-123) - (10,-122)
90 LINE (10,-122) - (200,-122) - (200,-121) -
     (10,-121) - (10,-120)
100 LINE (10,-120) - (10,-105) - (200,-105) -
     (200,-104) - (10,-104) - (10,-103) - (200,-103)
110 LINE (200,-103) - (10,-103) - (10,-102) -
     (200,-102) - (200,-101) - (10,-101) - (10,-100)
120 LINE (10,-100) - (200,-100) - (200,-99) -
     (10,-99) - (10,-98) - (200,-98) - (200,-97)
130 LINE (200,-97) - (10,-97) - (10,-96) - (200,-96)
     - (200,-95) - (10,-95) - (10,-94)
140 LINE (10,-94) - (200,-94) - (200,-93) - (10,-93)
     - (10,-92) - (200,-92) - (200,-91)
150 LINE (200,-91) - (10,-91) - (10,-90) - (200,-90)
     - (200,-75) - (10,-75) - (10,-74)
160 GLCURSOR (55,-74)
     : LINE (55,-74) - (200,-74) - (200,-73) -
     (55,-73) - (55,-72)
170 LINE (55,-72) - (200,-72) - (200,-71) - (55,-71)
     - (55,-70) - (200,-70) - (200,-69)
180 LINE (200,-69) - (55,-69) - (55,-68) - (200,-68)
     - (200,-67) - (55,-67) - (55,-66)
190 LINE (55,-66) - (200,-66) - (200,-65) - (55,-65)
     - (55,-64) - (200,-64) - (200,-63)
200 LINE (200,-63) - (55,-63) - (55,-62) - (200,-62)
     - (200,-61) - (55,-61) - (55,-60)
210 LINE (55,-60) - (200,-60)
     : GLCURSOR (55,-59)
220 LINE (55,-59) - (85,-59) - (85,-58) - (55,-58) -
     (55,-57) - (85,-57) - (85,-56)
230 LINE (85,-56) - (55,-56) - (55,-55) - (85,-55) -
     (85,-54) - (55,-54) - (55,-53)
240 LINE (55,-53) - (85,-53) - (85,-52) - (55,-52) -
     (55,-51) - (85,-51) - (85,-50)
250 LINE (85,-50) - (55,-50) - (55,-49) - (85,-49) -
     (85,-48) - (55,-48) - (55,-47)
260 LINE (55,-47) - (85,-47) - (85,-46) - (55,-46) -
     (55,-45) - (200,-45) - (200,-44)
270 LINE (200,-44) - (85,-44) - (85,-43) - (200,-43)
     - (200,-42) - (85,-42) - (85,-41)
280 LINE (85,-41) - (200,-41) - (200,-40) - (85,-40)
     - (85,-39) - (200,-39) - (200,-38)
290 LINE (200,-38) - (85,-38) - (85,-37) - (200,-37)
     - (200,-36) - (85,-36) - (85,-35)
300 LINE (85,-35) - (200,-35) - (200,-34) - (85,-34)
     - (85,-33) - (200,-33) - (200,-32)
310 LINE (200,-32) - (85,-32) - (85,-31) - (200,-31)
     - (200,-30) - (85,-30) - (85,-29)
320 LINE (55,-29) - (85,-29) - (85,-28) - (55,-28) -
     (55,-27) - (85,-27) - (85,-26)
330 LINE (85,-26) - (55,-26) - (55,-25) - (85,-25) -
     (85,-24) - (55,-24) - (55,-23)
340 LINE (55,-23) - (85,-23) - (85,-22) - (55,-22) -
     (55,-21) - (85,-21) - (85,-20)
350 LINE (85,-20) - (55,-20) - (55,-19) - (85,-19) -
     (85,-18) - (55,-18) - (55,-17)
360 LINE (55,-17) - (85,-17) - (85,-16) - (55,-16) -
     (55,-15) - (200,-15) - (200,-14)
370 LINE (200,-14) - (55,-14) - (55,-13) - (200,-13)
     - (200,-12) - (55,-12) - (55,-11)
380 LINE (55,-11) - (200,-11) - (200,-10) - (55,-10)
     - (55,-9) - (200,-9) - (200,-8)
390 LINE (200,-8) - (55,-8) - (55,-7) - (200,-7) -
     (200,-6) - (55,-6) - (55,-5)
400 LINE (55,-5) - (200,-5) - (200,-4) - (55,-4) -
     (55,-3) - (200,-3) - (200,-2)
410 LINE (200,-2) - (55,-2) - (55,-1) - (200,-1)
420 GLCURSOR (10,-75)
     : LINE (10,-75) - (40,-75) - (40,-74) - (10,-74)
     - (10,-73)
430 LINE (10,-73) - (40,-73) - (40,-72) - (10,-72) -
     (10,-71) - (40,-71) - (40,-70)
440 LINE (40,-70) - (10,-70) - (10,-69) - (40,-69) -
     (40,-68) - (10,-68) - (10,-67)
450 LINE (10,-67) - (40,-67) - (40,-66) - (10,-66) -
     (10,-65) - (40,-65) - (40,-64)
460 LINE (40,-64) - (10,-64) - (10,-63) - (40,-63) -
     (40,-62) - (10,-62) - (10,-61)
470 LINE (10,-61) - (40,-61) - (40,-60) - (10,-60) -
     (10,-59) - (40,-59)
480 LINE (40,-58) - (10,-58) - (10,-57) - (40,-57) -
     (40,-56) - (10,-56) - (10,-55)
490 LINE (10, - 55) - (40,-55) - (40,-54) - (10,-54)
     - (10,-53) - (40,-53) - (40,-52)
500 LINE (40,-52) - (10,-52) - (10,-51) - (40,-51) -
     (40,-50) - (10,-50) - (10,-49)
510 LINE (10,-49) - (40,-49) - (40,-48) - (10,-48) -
     (10,-47) - (40,-47) - (40,-46)
520 LINE (40,-46) - (10,-46) - (10,-45) - (40,-45)
521 GLCURSOR (40,-29)
522 LINE (40,-29) - (10,-29) - (10,-28) - (40,-28) -
     (40,-27) - (10,-27) - (10,-26)
540 LINE (10,-26) - (40,-26) - (40,-25) - (10,-25) -
     (10,-24) - (40,-24) - (40,-23)
550 LINE (40,-23) - (10,-23) - (10,-22) - (40,-22) -
     (40,-21) - (10,-21) - (10,-20)
560 LINE (10,-20) - (40,-20) - (40,-19) - (10,-19) -
     (10,-18) - (40,-18) - (40,-17)
570 LINE (40,-17) - (10,-17) - (10,-16) - (40,-16) -
     (40,-15) - (10,-15) - (10,-14)
580 LINE (10,-14) - (40,-14) - (40,-13) - (10,-13) -
     (10,-12) - (40,-12) - (40,-11)
590 LINE (40,-11) - (10,-11) - (10,-10) - (40,-10) -
     (40,-9) - (10,-9) - (10,-8)
600 LINE (10,-8) - (40,-8) - (40,-7) - (10,-7) -
     (10,-6) - (40,-6) - (40,-5)
610 LINE (40,-5) - (10,-5) - (10,-4) - (40,-4) -
     (40,-3) - (10,-3) - (40,-2)
620 LINE (10,-2) - (40,-2) - (40,-1) - (10,-1) -
     (10,0)
621 COLOR 0
622 GLCURSOR (5,5)
     : LINE (5,5) - (5,-270) - (0,-270) - (0,5) -
     (5,5)
630 TEXT
     : CSIZE 2
     : LF 10
     : LPRINT "  FLAG OF GREECE"
640 TEXT
     : LF 5
```

Document Listing for the Model II

Jean-Pierre Radley
320 Central Park West
New York, NY 10025

This program was written in interpreter BASIC and will list a BASIC program which has been saved in the normal format (not ASCII) in an attractive format.

```
Ø REM DOCLIST3/BAS Based on Lewis Rosenfelder, "Basic
     Faster & Better & Other Mysteries". Version
     September 1982 by Jean-Pierre Radley, 32Ø
     C.P.W., N.Y. 1ØØ25. Includes, in array format,
     assembly language for four USR calls.
1 CLEAR 1ØØØØ
     : DEFINT A-Y
     : DEFSTR Z
3 DIM B(1),RW$(128),US(45)
4 ZZ = CHR$(26)+" "
     : ZY = " "+CHR$(25)
     : ZX = CHR$(24)
5 PW = 8Ø
1Ø DEF FN DT$ = MID$(DATE$,1,3)+" "+MID$(DATE$,4,3)+"
     "+MID$(DATE$,7,2)+" "+MID$(DATE$,9,4)+" @
     "+TIME$
15 CLS
     : PRINT@85,"DOCLIST3/BAS will list in a
     "CHR$(34)"pretty-printing"CHR$(34)" format any
     BASIC program. The program must be on disk in
     normal format (i.e., not in ASCII format).
16 PRINT
     : PRINT"This program modifies the BASIC
     interpreter by replacing the OCT$ function by
     PEEK, and the NAME function by POKE. Re-entering
     BASIC from TRSDOS READY will restore normal
     BASIC operation.
2Ø ON ERROR GOTO 115Ø
     : CLOSE
     : OPEN"D",2,"DOCLIST3/BAS"
     : ON ERROR GOTO Ø
     : CLOSE
3Ø GOSUB 5ØØØ
5Ø PRINT@96Ø,"Loading reserved words...";
51 X = 1Ø323
     : Y = Ø
     : C = Ø
     : DEF USR6 = VARPTR(US(32))
     : REM peek routine starts at element 32 of
     US()52 C = USR6(X)
     : IF C > 127 THEN PRINT@992,Y,RW$(Y),
     : Y = Y+1
     : IF Y > 128 THEN 55 ELSE RW$(Y) = CHR$(C AND
     NOT 128)
     : GOTO 54
53 RW$(Y) = RW$(Y)+CHR$(C)
54 X = X+1
     : GOTO 52
55 RW$(13) = RW$(13)+"  ":
     REM make "IF"  4 characters long
56 RW$(2) = RW$(2)+" ":
     REM make "FOR" 4 characters long
1ØØ GOSUB 1ØØØ
11Ø GOSUB 11ØØ
12Ø GOSUB 12ØØ
13Ø CLS
     : PRINT PN$
14Ø GOSUB 21ØØ
15Ø PN = 1
     : GOSUB 3ØØØ
16Ø GOSUB 4ØØØ
17Ø IF INSTR(OP$,"P") = Ø THEN LPRINT STRING$(2,12)
171 IF INSTR(OP$,"D") THEN PRINT#2,STRING$(255,Ø)
18Ø CLOSE
     : PRINT"Press "ZZ"<F1>"ZY" for another run,
     "ZZ"<F2>"ZY" to end..."
     : A$ = INPUT$(1)
     : IF A$ = CHR$(1) THEN CLS
     : GOTO 1ØØ ELSE IF A$ = CHR$(2) THEN PRINT
     TAB(13Ø) "Thanks"
     : END
     : ELSE 18Ø
1ØØØ REM Initialize simple variables
1Ø1Ø C = Ø
     : P = Ø
     : BP = Ø
     : PC = Ø
     : LN$ = ""
     : VB = Ø
     : NF = Ø
     : FF = Ø
     : NT = Ø
     : FX$ = ""
     : QF = Ø
     : I1 = 5
     : I2 = 5
     : FL! = Ø
     : TL! = 65536
     : RN = 1
1Ø2Ø RETURN
11ØØ REM Enter program name, open and field program
     file
1111Ø PRINT@32Ø,ZX "What is the name of the program to
     be listed ? ";
1111 LINEINPUT PN$
1112 ON ERROR GOTO 114Ø
     : CLOSE
     : OPEN"I",1,PN$
     : CLOSE
     : OPEN"D",1,PN$
     : ON ERROR GOTO Ø
1115 J = Ø
     : DEFUSR6 = VARPTR(US(Ø))
     : REM poke routine starts at element Ø of US()
     and is fed Addr,Byte. If reassembling, note that
     USR6 address (2B36) is specifically referenced
     in the routine
112Ø FIELD 1,128 AS B$(Ø),127 AS B$(1)
     : J = USR6(VARPTR(B$(1))) OR USR6(128)
113Ø RETURN
1141Ø IF ERR = 54 THEN RESUME NEXT ELSE GOSUB 115Ø
     : RESUME 11ØØ
115Ø REM Handling errors resulting from opening
     program files
1151 PRINT
     : PRINT
     : PRINT
     : IF ERR = 53 THEN PRINT "File not found" ELSE
     IF ERR = 52 THEN PRINT "Bad file number -
     re-entering BASIC with -F:2"
     : CLOSE
     : SYSTEM"BASIC DOCLIST3/BAS -F:2" ELSE
     PRINT"Error "ZZ;ERR;ZY" in Line "ZZ;ERL;ZY
1152 PRINT"Press any key...";A$ = INPUT$(1)
     : RETURN
12ØØ REM Select options
12Ø5 CLS
     : PRINT@(2,27),CHR$(26)" "PN$" "CHR$(25)
121Ø PRINT@4ØØ,ZZ"R"ZY" Line number
     range";TAB(4Ø)ZZ"W"ZY" Page width <>
     8Ø";TAB(16Ø)ZZ"S"ZY" Stop after each
     page";TAB(4Ø)ZZ"D"ZY" Output to
     disk";TAB(16Ø)ZZ"H"ZY" Special page
     heading";TAB(4Ø)ZZ"P"ZY" No output to printer"
1215 PRINT STRING$(2,13)"Type (in any order) the
     letters corresponding to the options you want,
```

```
     if any, and "ZZ"<ENTER>"ZY
     : PRINT
 122Ø LINEINPUT OP$
 123Ø IF INSTR(OP$,"R") = Ø THEN 124Ø ELSE
     PRINT@128Ø,CHR$(24);
 1231 INPUT"From line ";FL!
 1232 INPUT" To line ";TL!
 124Ø IF INSTR(OP$,"D") = Ø THEN 125Ø ELSE
     PRINT@144Ø,CHR$(24);
 1241 LINEINPUT"Output disk file name: ";A$
     : IF A$ = PN$ THEN PRINT@1491,"Cannot use "PN$
     : GOTO 1241
 1242 OPEN "O",2,A$
 125Ø IF INSTR(OP$,"W") = Ø THEN 126Ø ELSE
     PRINT@152Ø,ZX;
 1251 INPUT"page width ";PW
 126Ø IF INSTR(OP$,"H") = Ø THEN 127Ø ELSE
     PRINT@16ØØ,ZX
 1261 PRINT"Enter the page heading (up
     to"PW"characters):"
     : LINEINPUT PH$
 1262 PH$ = SPACE$((PW-LEN(PH$))\2)+PH$
 127Ø RETURN
 2ØØØ REM Get next byte from disk file - return as C%
 2Ø1Ø P = P+1
     : IF P < 129 THEN 2Ø2Ø ELSE P = 1
     : BP = BP+1
     : IF BP < 2 THEN 2Ø2Ø ELSE BP = Ø
     : GOSUB 21ØØ
 2Ø2Ø C = ASC(MID$(B$(BP),P))
     : RETURN
 21ØØ REM Get next record from disk file
 211Ø GET 1,RN
     : RN = RN+1
 212Ø RETURN
 22ØØ REM Get next 2 bytes from disk file - return as
     A!
 221Ø GOSUB 2ØØØ
     : PC = C
     : GOSUB 2ØØØ
     : A! = CVI(CHR$(PC)+CHR$(C))
     : IF A! < Ø THEN A! = 65536+A!
 222Ø RETURN
 3ØØØ REM Prepare printer
 3Ø1Ø IF INSTR(OP$,"P") THEN RETURN
 3Ø2Ø PW$ = "FORMS W = "+STR$(PW)
     : SYSTEM PW$
 3Ø3Ø J = Ø
     : DEFUSR6 = VARPTR(US(42))
     : J = USR6(Ø)
     : REM routine starting at element 42 of US()
     will set line count to zero
 31ØØ REM Print page heading
 311Ø IF INSTR(OP$,"P") THEN RETURN
 3115 ON ERROR GOTO 316Ø
 312Ø ZS = STRING$((PW-35-LEN(PN$))\2,32)
     : LPRINT CHR$(34)PN$CHR$(34)ZS FNDT$ZS"page"PN
 3125 ON ERROR GOTO Ø
 313Ø IF INSTR(OP$,"H") THEN LPRINT PH$
 314Ø LPRINT STRING$(PW,"=")
     : LPRINT" "
 315Ø PN = PN+1
     : RETURN
 316Ø PRINT"Error"ZZ;ERR;ZY"in line"ZZ;ERL;ZY". Are
     you sure the printer is ready?"
     : ZQ = INPUT$(1)
     : RESUME 3Ø2Ø
 32ØØ REM Print a line of text
 321Ø PRINT LN$
 3211 IF INSTR(OP$,"P") = Ø THEN LPRINT LN$;
     : GOSUB 44ØØ
 3212 IF INSTR(OP$,"D") THEN PRINT#2,LN$;
 322Ø IF INSTR(" 128 138 139 143 158 165 171 183
     ",STR$(VB)) = Ø OR (PC <> 58 AND C <> Ø) THEN
     324Ø
 323Ø IF FF + NF = Ø THEN 3235 ELSE NT = NT+1
 3231 IF INSTR(OP$,"P") = Ø THEN LPRINT"
     "STRING$(PW-LEN(LN$)-2,".");
     : GOSUB 44ØØ
 3232 IF INSTR(OP$,"D") THEN PRINT#2,"
     "STRING$(PW-LEN(LN$)-2,".");
 3233 IF (C = Ø) AND (NT/2 <> INT(NT/2)) THEN 3235
     ELSE 324Ø
 3235 IF INSTR(OP$,"P") = Ø THEN LPRINT" "
     : GOSUB 44ØØ
     : LPRINT STRING$(PW-1,"-");
     : GOSUB 44ØØ
 3236 IF INSTR(OP$,"D") THEN PRINT#2," "
     : PRINT#2,STRING$(PW-1,"-");
 324Ø IF INSTR(OP$,"P") = Ø THEN LPRINT" "
     : GOSUB 44ØØ
 3241 IF INSTR(OP$,"D") THEN PRINT#2," "
 325Ø LN$ = STRING$(6+NF+FF," ")
     : RETURN
 33ØØ REM Test on print-line length - print if filled
 331Ø IF LEN(LN$)+6 < PW THEN RETURN ELSE GOSUB 32ØØ
     : RETURN
 4ØØØ REM Process the text
 4Ø1Ø GOSUB 22ØØ
     : IF A! = Ø THEN 4Ø4Ø
 4Ø2Ø GOSUB 22ØØ
     : IF A! < FL! THEN PRINT A!
     : GOSUB 43ØØ
     : GOTO 4Ø1Ø ELSE IF A! > TL! THEN 4Ø4Ø
 4Ø3Ø GOSUB 41ØØ
     : GOSUB 32ØØ
     : GOTO 4Ø1Ø
 4Ø4Ø FF = Ø
     : NF = Ø
     : C = 1
     : GOSUB 32ØØ
     : RETURN
 41ØØ REM Process a line
 411Ø QF = Ø
     : FF = Ø
     : FX$ = ""
     : C = Ø
     : VB = Ø
     : NT = Ø
 412Ø LN$ = RIGHT$("     "+STR$(A!),5)+"
     "+STRING$(NF," ")
 413Ø PC = C
     : GOSUB 2ØØØ
     : IF C = Ø THEN RETURN
 4135 IF C = 146 THEN GOSUB 32Ø0
     : MID$(LN$,LEN(LN$)-4,4) = "ELSE"
     : VB = 138
     : IF FX$ = "ELSE" THEN LN$ = MID$(LN$,I1+1)
     : FF = (FF-I1)*-(I1<=FF)
     : GOTO 413Ø ELSE FX$ = "ELSE"
     : GOTO 413Ø
 414Ø IF PC = 58 AND QF = Ø AND VB <> Ø THEN GOSUB
     32ØØ
 415Ø IF C > 127 THEN 418Ø
 416Ø IF C = 34 THEN QF = NOT QF
 4161 IF (C =1Ø AND QF = Ø) OR (C = 32 AND QF = Ø)
     THEN 413Ø
 4162 IF C = 1Ø THEN GOSUB 32ØØ
     : GOTO 413Ø
 417Ø LN$ = LN$+CHR$(C)
     : GOSUB 33ØØ
     : GOTO 413Ø
 418Ø REM Process reserved word
 4182 IF C = 199 THEN GOSUB 32ØØ
     : MID$(LN$,LEN(LN$)-4,4) = "THEN"
     : VB = 138
     : GOTO 413Ø
 4184 IF C = 132 AND FX$ = "" THEN
     MID$(LN$,LEN(LN$)-4,4) = "NEXT"
     : NF = (NF-I2)*-(I2<=NF)
     : GOTO 413Ø
 4186 IF C = 14Ø THEN FF = FF + I1
     : NT = NT+1
```

```
     : FX$ = "IF"
4188 IF C = 129 THEN NF = NF+12
4190 IF C = 144 THEN QF = -2
     : IF PC = 58 THEN MID$(LN$,LEN(LN$),1)="'"
     : GOSUB 3300
     : GOTO 4130
4200 IF RIGHT$(LN$,1) <> " " THEN LN$ = LN$+" "
4201 LN$ = LN$+RW$(C-127)+" "
     : GOSUB 3300
4210 IF C = 138 AND VB = 157 THEN VB = -1
     : GOTO 4130 ELSE VB = C
     : GOTO 4130
4300 REM Read to end of text line - ignoring contents
4310 GOSUB 2000
     : IF C = 0 THEN RETURN
4320 P = INSTR(P,B$(BP),CHR$(0))
     : IF P > 0 THEN C = 0
     : RETURN ELSE P = 128
     : GOTO 4310
4400 REM End of page routine
4401 DEFUSR6 = VARPTR(US(37))
     : J = USR6(0)
     : REM routine starting at element 37 of US()will
     return in J the # of lines printed since last
     top-of-form
4402 IF J <> 0 THEN RETURN ELSE IF INSTR(OP$,"S")
     THEN PRINT
     : PRINT"Press "ZZ"F1"ZY" to continue, "ZZ"F2"ZY"
     to stop..."
     : A$ = INPUT$(1)
     : IF A$ = CHR$(2) THEN CLOSE
     : END ELSE IF A$ <> CHR$(1) THEN 4402
4403 GOSUB 3100
     : RETURN
5000 REM Magic array for peeking, poking, counting
     number of lines since last top-of-page on the
     printer, setting this count to zero
5010 FOR X = 0 TO 45
     : READ US(X)
     : NEXT
5020 DATA 24013, -8892, 13866, -8917, 12405, 29917,
     -8911, 2356
5021 DATA 13533, -8951, 3124, 13533, -8948, 2430,
     12294, -8816
5022 DATA 12102, 10384, -14079, 14045, 12297, 14045,
     12556, 1048
5023 DATA 11264, 65, 28381, -8912, 12646, 32477,
     30514, 201
5024 DATA 24013, 28228, 38, 31427, 68, 24382, 6, 9935
5025 DATA -15616, 17530, 24382, 1, -12541, 201
5030 RETURN
```

A Tribute to Columbia

Arnold E. van Beverhoudt, Jr.
P.O. Box 56
St. Thomas, V.I. 00801

In November 1982, the space shuttle began its operational flights, opening up space for practical use. The space shuttle is a great achievement of which I, as an American and a supporter of NASA and the space program, am proud. My program is a tribute to the men and women of NASA and to the space shuttle, particularly the first one to fly, "Columbia." The program also shows the kind of detailed graphics that are possible on the "low resolution" Models I and III. A Tribute to Columbia also includes a screen print routine which shows off the capabilities of the Line Printer VII.

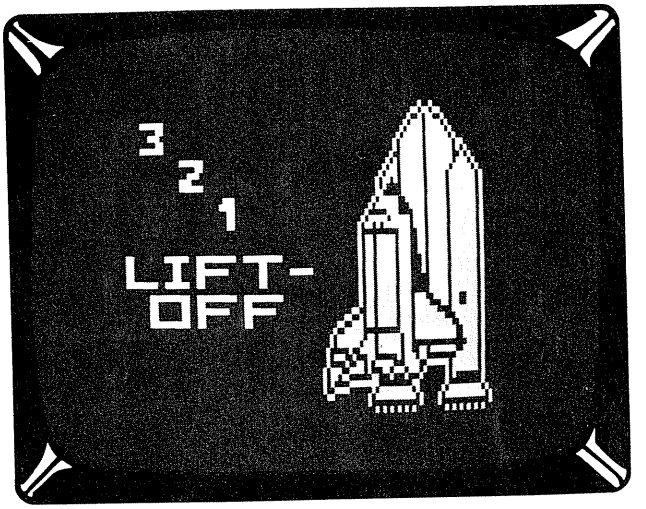
I am the proud owner of a 48K Model I with one disk and an LP VII. The good old Model I has stood up great to almost three years of daily use by my wife, my 5-year old daughter, and myself. It is an important practical, educational, and recreational part of our household.

The BASIC program, as shown, requires 32K to run. However, 16K users can have the graphic displays by deleting the narrative descriptions in lines 350-1150, the screen-print routine in lines 39999-50000, the GOSUB 40000 in lines 295, 325, 345, and all REM statements.

```
10  REM ***********************************
    ***** A TRIBUTE TO 'COLUMBIA' *****
    ************ (1982) BY ************
** ARNOLD E. VAN BEVERHOUDT, JR. **
    ***********************************
100 REM ***** INITIALIZATION ROUTINES *****
110 CLEAR 500
    : DIM H$(15)
    : CLS
    : PRINT CHR$(23)
    : PRINT
    : PRINT
    : PRINT
    : PRINT TAB(5);"A TRIBUTE TO 'COLUMBIA'"
120 PRINT
    : PRINT
    : PRINT
    : PRINT TAB(15);"BY"
    : PRINT
    : PRINT
    : PRINT
130 PRINT TAB(2); "ARNOLD E. VAN BEVERHOUDT, JR."
    : FOR T = 1 TO 1000
    : NEXT T
140 H$(1) = CHR$(131) + CHR$(179) + CHR$(159)
    : H$(2) = STRING$(2,140) + CHR$(143)
150 H$(3) = CHR$(131) + CHR$(179) + CHR$(159)
    : H$(4) = CHR$(143) + CHR$(141) + CHR$(140)
160 H$(5) = CHR$(142) + CHR$(191)
    : H$(6) = CHR$(128) + CHR$(143)
170 H$(7) = CHR$(191) + STRING$(4,128) + CHR$(131) +
    CHR$(191) + CHR$(131) + CHR$(128) + CHR$(191) +
    STRING$(3,179) + CHR$(131) + CHR$(128) +
    STRING$(2,131) + CHR$(191) + STRING$(2,131) +
    CHR$(128) + STRING$(3,176)
180 H$(8) = CHR$(143) + STRING$(3,140) + CHR$(128) +
    CHR$(140) + CHR$(143) + CHR$(140) + CHR$(128) +
    CHR$(143) + STRING$(7,128) + CHR$(143)
190 H$(9) = CHR$(191) + STRING$(3,131) + CHR$(191) +
    CHR$(128) + CHR$(191) + STRING$(3,179) +
    CHR$(131) + CHR$(128) + CHR$(191) +
    STRING$(3,179) + CHR$(131)
200 H$(10) = CHR$(143) + STRING$(3,140) + CHR$(143) +
    CHR$(128) + CHR$(143) + STRING$(5,128) +
    CHR$(143)
210 H$(11) = STRING$(6,149) + STRING$(3,128) +
    STRING$(6,170)
220 H$(12) = STRING$(2,131) + CHR$(191) +
    STRING$(2,131) + CHR$(128) + CHR$(191) +
    STRING$(3,131) + CHR$(191) + CHR$(128) +
    CHR$(191) + STRING$(3,128) + CHR$(191) +
    CHR$(128) + CHR$(191) + STRING$(4,131) +
    CHR$(128) + CHR$(191) + STRING$(3,176) +
    CHR$(191) + CHR$(128) + CHR$(191)
230 H$(13) = CHR$(131) + CHR$(139) + CHR$(180) +
    CHR$(128) + CHR$(191) + STRING$(3,131) +
    CHR$(191) + CHR$(128) + CHR$(191) + CHR$(128) +
    CHR$(188) + CHR$(128) + CHR$(191) + CHR$(128) +
    STRING$(2,191) + CHR$(176) + CHR$(128) +
    CHR$(191) + STRING$(2,128) + CHR$(191)
240 H$(14) = CHR$(143) + STRING$(3,128) + CHR$(143) +
    STRING$(3,140) + CHR$(143) + CHR$(128) +
    CHR$(143) + STRING$(3,140) + CHR$(143) +
```

```
      CHR$(128) + CHR$(143) + STRING$(4,14Ø) +
      CHR$(128) + CHR$(143) + STRING$(3,128) +
      CHR$(143) + CHR$(128) + CHR$(143) + CHR$(14Ø) +
      CHR$(142) + CHR$(129)
25Ø H$(15) = CHR$(128) + CHR$(143) + STRING$(3,14Ø) +
      CHR$(143) + CHR$(128) + CHR$(143) + CHR$(131) +
      CHR$(128) + CHR$(131) + CHR$(143) + CHR$(128) +
      CHR$(143) + CHR$(128) + CHR$(131) +
      STRING$(2,143) + STRING$(2,128) + CHR$(14Ø)
255 REM ***** PRINT GRAPHICS ROUTINES *****
26Ø RESTORE
      : CLS
      : FOR T = 15360 TO 16383
      : READ G
      : POKE T,G
      : NEXT T
      : FOR T = 1 TO 15ØØ
      : NEXT T
27Ø PRINT@67,H$(1);
      : PRINT@131,H$(2);
      : FOR T = 1 TO 5ØØ
      : NEXT T
      : PRINT@2ØØ,H$(3);
      : PRINT@264,H$(4);
28Ø FOR T = 1 TO 5ØØ
      : NEXT T
      : PRINT@333,H$(5);
      : PRINT@397,H$(6);
      : FOR T = 1 TO 5ØØ
      : NEXT T
29Ø PRINT@513,H$(7);
      : PRINT@577,H$(8);
      : PRINT@644,H$(9);
      : PRINT@7Ø8,H$(1Ø);
      : FOR T = 1 TO 1ØØØ
      : NEXT T
295 FOR T = 1 TO 1ØØØ
      : GOSUB 4ØØØØ
      : NEXT T
3ØØ FOR T = Ø TO 14
      : PRINT@Ø + 64 * T,STRING$(25,128);
      : NEXT T
31Ø PRINT@96Ø,"";
      : FOR T = 1 TO 1Ø
      : PRINT TAB(33);H$(11)
      : FOR TT = 1 TO 1ØØ
      : NEXT TT
      : NEXT T
      : FOR T =1 TO 2Ø
      : PRINT
      : FOR TT = 1 TO 1ØØ
      : NEXT TT
      : NEXT T
32Ø FOR T = 15360 TO 16383
      : READ G
      : POKE T,G
      : NEXT T
      : PRINT@3Ø,">> IN ORBIT <<";
325 FOR T = 1 TO 1ØØØ
      : GOSUB 4ØØØØ
      : NEXT T
33Ø CLS
      : FOR T = 15360 TO 16383
      : READ G
      : POKE T,G
      : NEXT T
      : FOR T = 1 TO 1ØØØ
      : NEXT T
34Ø PRINT@69,H$(12);H$(13);
      : PRINT@135,H$(14);H$(15);
345 FOR T = 1 TO 1ØØØ
      : GOSUB 4ØØØØ
      : NEXT T
35Ø REM ***** PRINT NARRATIVE ROUTINE *****
36Ø CLS
      : PRINTTAB(16);"*** A TRIBUTE TO 'COLUMBIA' ***"
      : PRINT
37Ø PRINT"THIS SHORT GRAPHIC PROGRAM IS A TRIBUTE TO
      N.A.S.A. AND THE"
38Ø PRINT"SPACE SHUTTLE - IN PARTICULAR THE
      'COLUMBIA', THE FIRST SHUTTLE"
39Ø PRINT"TO FLY."
      : PRINT
4ØØ PRINT"THE OBJECTIVES OF THE SPACE SHUTTLE PROGRAM
      ARE:"
41Ø PRINT"- TO ESTABLISH A NATIONAL SPACE
      TRANSPORTATION PROGRAM THAT"
42Ø PRINT"      WILL SUBSTANTIALLY REDUCE THE COST OF
      SPACE OPERATIONS."
43Ø PRINT"- TO PROVIDE A CAPABILITY TO SUPPORT A WIDE
      RANGE OF SCIENTIFIC"
44Ø PRINT"      APPLICATIONS & COMMERCIAL, DEFENSE, &
      INTERNATIONAL USES."
45Ø PRINT
      : PRINT"THE SPACE SHUTTLE IS COMPOSED OF THREE
      BASIC ELEMENTS:"
46Ø PRINT TAB(5);"- THE SHUTTLE ORBITER (122 FT. LONG
      - 78 FT. WINGSPAN)"
47Ø PRINTTAB(5);"- AN EXTERNAL FUEL TANK (1.6 MILLION
      LBS. CAPACITY)"
48Ø PRINTTAB(5);"- TWO SOLID ROCKET BOOSTERS (2.6
      MILLION LBS. THRUST EACH)";
      : GOSUB 5ØØØØ
49Ø CLS
      : PRINT"THE SPACE SHUTTLE IS THE WORLD'S FIRST
      REUSABLE SPACECRAFT."
5ØØ PRINT"IT IS LAUNCHED INTO SPACE AS A CONVENTIONAL
      ROCKET, AND RETURNS"
51Ø PRINT"TO EARTH AND LANDS LIKE AN AIRPLANE. DURING
      REGULAR FLIGHTS,"
52Ø PRINT"THE SHUTTLE WILL BE LAUNCHED FROM THE SAME
      LAUNCH PADS AT THE"
53Ø PRINT"KENNEDY SPACE CENTER IN FLORIDA THAT
      LAUNCHED THE APOLLO SPACE-"
54Ø PRINT"CRAFTS, AND IT WILL LAND ON A 3-MILE LONE
      CONCRETE RUNWAY AT"
55Ø PRINT"THE KENNEDY SPACE CENTER. SOME MILITARY
      FLIGHTS OF THE SPACE"
56Ø PRINT"SHUTTLE WILL BEGIN AND END AT A NEW SPACE
      CENTER AT VANDENBERG"
57Ø PRINT"AIR FORCE BASE IN CALIFORNIA. THE
      VANDENBERG LAUNCH SITE WILL"
58Ø PRINT"ALLOW FLIGHT INTO ORBITS WITH A HIGHER
      INCLINATION THAN KENNEDY"
59Ø PRINT"SPACE CENTER LAUNCHES."
      : PRINT
6ØØ PRINT"IN ITS CARGO BAY, THE SHUTTLE CAN CARRY
      PAYLOADS OF UP TO"
61Ø PRINT"65,ØØØ LBS. AND 6Ø FT. LONG. A
      REMOTE-CONTROLLED ARM IN THE"
62Ø PRINT"CARGO BAY WILL BE USED TO REMOVE AND
      REPLACE SATELLITES AND"
63Ø PRINT"OTHER EXPERIMENT PACKAGES TAKEN INTO
      SPACE.";
      : GOSUB 5ØØØØ
64Ø CLS
      : PRINT"ON SOME MISSIONS OF THE SPACE SHUTTLE,
      IT WILL CARRY A SPACE"
65Ø PRINT"LABORATORY, SIMILAR TO SKYLAB, IN ITS CARGO
      BAY. THIS LAB IS"
66Ø PRINT"BUILT BY THE EUROPEAN SPACE AGENCY
      (E.S.A.), AND CREWS OF UP TO"
67Ø PRINT"SIX ASTRONAUTS (AMERICAN AND EUROPEAN) WILL
      CARRY OUT EXPERI-"
68Ø PRINT"MENTS FOR UP TO TWO WEEKS IN DURATION."
      : PRINT
69Ø PRINT"THE SPACE SHUTTLE WILL ALSO BE USED TO
      CARRY MANY SCIENTIFIC,"
7ØØ PRINT"DEFENSE, COMMUNICATIONS, AND WEATHER
      SATELLITES INTO ORBIT."
71Ø PRINT"IT WILL ALSO RETRIEVE BROKEN SATELLITES AND
      BRING THEM BACK TO"
72Ø PRINT"EARTH FOR REPAIR AND RE-LAUNCH."
      : PRINT
```

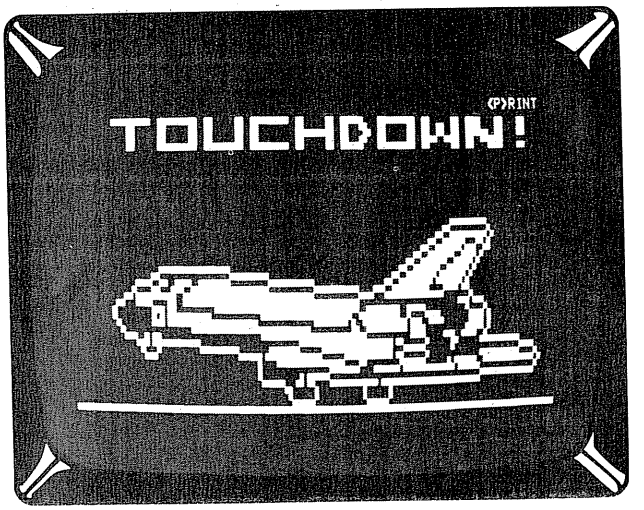
```
73Ø PRINT"IN ALL, THERE WILL BE FIVE SPACE SHUTTLES
    BUILT:"
74Ø PRINT"'ENTERPRISE' -- USED FOR LANDING TESTS;
    WILL NOT GO INTO SPACE"
75Ø PRINT"'COLUMBIA'   -- THE FIRST SHUTTLE TO
    ACTUALLY FLY IN SPACE"
76Ø PRINT"'CHALLENGER'              'DISCOVERY'
    'ATLANTIS'";
    : GOSUB 5ØØØØ
77Ø CLS
    : PRINT TAB(2Ø);"SPACE SHUTTLE FLIGHT LOG"
    : PRINT
78Ø PRINT"** ENTERPRISE **"
    : PRINT
79Ø PRINT" 6/18/77  FIRST MANNED FLIGHT ATTACHED TO
    THE TOP OF A 747"
8ØØ PRINT"          ASTRONAUTS FRED HAISE & GORDON
    FULLERTON"
81Ø PRINT" 6/28/77  SECOND MANNED FLIGHT ATTACHED TO
    THE TOP OF A 747"
82Ø PRINT"          ASTRONAUTS JOE ENGLE & RICHARD
    TRULY"
    : PRINT
83Ø PRINT" 8/12/77  FIRST MANNED FREE FLIGHT TO A
    LANDING AT EDWARDS AFB"
84Ø PRINT"          ASTRONAUTS FRED HAISE & GORDON
    FULLERTON"
85Ø PRINT" 8/3Ø/77  SECOND MANNED FREE FLIGHT TO A
    LANDING AT EDWARDS AFB"
86Ø PRINT"          ASTRONAUTS JOE ENGLE & RICHARD
    TRULY"
87Ø PRINT" 9/16/77  THIRD MANNED FREE FLIGHT TO A
    LANDING AT EDWARDS AFB"
88Ø PRINT"          ASTRONAUTS FRED HAISE & GORDON
    FULLERTON";
    : GOSUB 5ØØØØ
89Ø CLS
    : PRINT TAB(2Ø);"SPACE SHUTTLE FLIGHT LOG"
    : PRINT
    : PRINT"** ENTERPRISE **  (CONT'D)"
    : PRINT
9ØØ PRINT"1Ø/13/77  FOURTH MANNED FREE FLIGHT TO A
    LANDING AT EDWARDS AFB"
91Ø PRINT"          ASTRONAUTS JOE ENGLE & RICHARD
    TRULY"
92Ø PRINT"1Ø/26/77  LAST MANNED FREE FLIGHT TO A
    LANDING AT EDWARDS AFB"
93Ø PRINT"          ASTRONAUTS FRED HAISE & GORDON
    FULLERTON"
94Ø PRINT
    : PRINT
    : PRINT"THE 'ENTERPRISE' WAS LATER USED IN
    STRESS AND VIBRATION TESTS";
    : GOSUB 5ØØØØ
95Ø CLS
    : PRINTTAB(2Ø);"SPACE SHUTTLE FLIGHT LOG"
    : PRINT
96Ø PRINT"** COLUMBIA **"
    : PRINT
97Ø PRINT" 4/12/81  STS-1  FIRST ORBITAL TEST - NO
    PAYLOAD"
98Ø PRINT"          2 DAYS  6 HRS.  22 MIN.  -  36
    ORBITS"
99Ø PRINT"          LANDED AT EDWARDS AFB, CA"
1ØØØ PRINT"         ASTRONAUTS JOHN YOUNG & ROBERT
    CRIPPEN"
    : PRINT
1Ø1Ø PRINT"11/12/81  STS-2  SECOND ORBITAL TEST -
    OSTA-1 EXPERIMENTS"
1Ø2Ø PRINT"          2 DAYS  6 HRS.  13 MIN.  -  36
    ORBITS"
1Ø3Ø PRINT"          LANDED 2 DAYS EARLY AT EDWARDS
    AFB, CA"
1Ø4Ø PRINT"          ASTRONAUTS JOE ENGLE & RICHARD
    TRULY";
    : GOSUB 5ØØØØ
1Ø5Ø CLS
    : PRINT TAB(2Ø);"SPACE SHUTTLE FLIGHT LOG"
    : PRINT
1Ø6Ø PRINT"** COLUMBIA **  (CONT'D)"
    : PRINT
1Ø7Ø PRINT" 3/22/82  STS-3  THIRD ORBITAL TEST -
    OSS-1 EXPERIMENTS"
1Ø8Ø PRINT"          8 DAYS  4 MIN.  -  159 ORBITS"
1Ø9Ø PRINT"          LANDED 1 DAY LATE AT WHITE SANDS
    MISSILE RANGE, NM"
11ØØ PRINT"          ASTRONAUTS JACK LOUSMA & GORDON
    FULLERTON"
    : PRINT
1111Ø PRINT" 6/27/82  STS-4  LAST ORBITAL TEST - DOD
    82-1 EXPERIMENTS"
112Ø PRINT"          7 DAYS  1 HR.  9 MIN.  -  112
    ORBITS"
113Ø PRINT"          LANDED ON CONCRETE RUNWAY AT
    EDWARDS AFB, CA"
1141Ø PRINT"         ASTRONAUTS TOM MATTINGLY & HENRY
    HARTSFIELD";
    : GOSUB 5ØØØØ
115Ø CLS
    : PRINT CHR$(23)
    : PRINT
    : PRINT
    : PRINT
    : PRINT TAB(6);"MORE FLIGHTS TO COME"
116Ø FOR T = 1 TO 15ØØ
    : NEXT T
    : GOTO 26Ø
```




```
1ØØØØ REM ***** LIFT-OFF GRAPHICS *****
1ØØ1Ø DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
1ØØ15 DATA 152, 166, 172, 153, 164, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
1ØØ2Ø DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 16Ø, 166
1ØØ25 DATA 19Ø, 191, 17Ø, 191, 189, 153, 144, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
1ØØ3Ø DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     16Ø, 133, 191
```

```
10035 DATA 191, 191, 170, 191, 191, 191, 138, 153,
     144, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10040 DATA 128,128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 152,
     185, 149, 191
10045 DATA 191, 191, 170, 191, 191, 167, 190, 191,
     182, 164, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10050 DATA 128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 170, 154,
     129, 129, 191
10055 DATa 191, 191, 170, 191, 191, 170, 191, 191,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10060 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 152, 137, 143,
     191, 149, 130
10065 DATA 175, 191, 170, 191, 191, 170, 191, 191,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10070 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 150, 142, 139, 139, 142,
     175, 191
10075 DATA 138, 191, 170, 191, 191, 170, 191, 191,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10080 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 149, 191, 170, 191,
     191, 170, 191
10085 DATA 128, 191, 170, 191, 191, 170, 191, 191,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10090 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 149, 191,
     170,191,191,170,191
10095 DATA 128, 170, 170, 191, 191, 170, 191, 191,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10100 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 149, 191, 170, 191,
     191, 170, 191
10105 DATA 128, 128, 170, 191, 191, 170, 191, 191,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10110 DATA 128,128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 176, 133, 191, 170, 191,
     191, 170, 191
10115 DATA 170, 191, 180, 147, 143, 170, 181, 186,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10120 DATA 128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 160, 140, 179, 188, 149, 143, 138, 143,
     143, 170, 191
10125 DATA 170, 191, 191, 191, 189, 180, 139, 191,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10130 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128 ,128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     160, 133, 143, 175, 143, 185, 191, 170, 191,
     191, 149, 191
10135 DATA 130, 179, 163, 179, 179, 179, 128, 187,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10140 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     154, 178, 147, 164, 185, 148, 151, 140, 188,
     164, 177, 175
10145 DATA 189, 163, 140, 140, 172, 168, 191, 191,
     191, 170, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10150 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     149, 167, 178, 153, 140, 140, 174, 165, 176,
     177, 176, 177
10155 DATA 172, 129, 128, 128, 154, 178, 176, 176,
     176, 154, 144, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
10160 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     130, 129, 128, 128, 128, 128, 133, 147, 147,
     147, 147, 147
10165 DATA 145, 133, 128, 138, 162, 163, 163, 163,
     163, 163, 138, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
20000 REM ***** IN ORBIT GRAPHICS *****
20010 DATA 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 188,
     176, 144, 128, 128, 128, 128, 129, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
20015 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
20020 DATA 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 188, 176, 128, 186, 135, 131,
     131, 131, 131, 131, 131, 131, 131, 131,
     143, 188, 176
20025 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 129, 160, 176, 176,
     176, 176, 176, 176, 128, 128, 128, 128, 129
20030 DATA 191, 191, 191, 191, 191, 191, 191, 159,
     143, 179, 179, 179, 188, 179, 143, 191, 191,
     191, 191, 191, 191, 191, 191, 188, 188, 176,
     128, 128, 128, 128, 128, 129, 128, 128, 128,
     128, 128, 131
20035 DATA 143, 188, 176, 128, 128, 128, 129, 128,
     176, 140, 179, 147, 179, 163, 177, 140, 188,
     188, 184, 140, 172, 147, 137, 140, 164, 128
20040 DATA 191, 191, 191, 191, 191, 191, 191, 191,
     180, 131, 188, 182, 143, 191, 191, 188, 179,
     143, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 189, 188, 176, 128, 128, 128, 128, 128,
     128, 176, 176
20045 DATA 140, 140, 131, 143, 188, 176, 140, 142,
     135, 168, 191, 191, 191, 191, 159, 133, 176,
     191, 191, 188, 143, 143, 129, 128, 160, 133
20050 DATA 191, 191, 191, 191, 191, 191, 191, 191,
     191, 181, 138, 191, 191, 176, 143, 191, 191,
     191, 176, 143, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 132, 140, 179, 179,
     179, 188, 188
20055 DATA 191, 191, 143, 143, 176, 179, 188, 191,
     189, 188, 140, 176, 176, 188, 188, 180, 139,
     191, 191, 149, 128, 128, 160, 140, 129, 128
20060 DATA 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 189, 146, 175, 191, 143, 128, 175,
     191, 191, 191, 188, 179, 143, 191, 191, 143,
     179, 179, 188, 188, 188, 191, 191, 143, 143,
```

```
    179, 179, 179
20065 DATA 188, 188, 191, 143, 143, 179, 179, 179,
     188, 188, 191, 191, 191, 143, 143, 179, 177,
     140, 167, 153, 140, 134, 129, 128, 128, 128
20070 DATA 191, 191, 191, 191, 191, 191, 191, 191,
     191, 159, 135, 145, 128, 140, 143, 143, 143,
     191, 143, 143, 179, 179, 188, 188, 188, 179,
     179, 143, 143, 179, 188, 188, 188, 191, 191,
     143, 143, 179
20075 DATA 179, 188, 188, 191, 191, 191, 191, 143,
     143, 179, 179, 188, 188, 191, 143, 179, 140,
     131, 129, 128, 128, 128, 128, 128, 128, 128
20080 DATA 179, 179, 179, 179, 179, 179, 179, 147,
     129, 142, 129, 128, 128, 130, 175, 191, 191,
     188, 147, 163, 139, 191, 191, 191, 191, 191,
     191, 191, 149, 138, 143, 179, 179, 188, 188,
     191, 191, 191
20085 DATA 191, 143, 143, 179, 179, 188, 188, 191,
     191, 191, 191, 143, 167, 153, 134, 129, 128,
     128, 128, 128, 128, 128, 129, 128, 128, 128
20090 DATA 188, 188, 188, 179, 179, 179, 179, 147,
     143, 191, 180, 128, 128, 128, 178, 179, 179,
     148, 136, 140, 140, 191, 191, 191, 191, 191,
     191, 191, 149, 170, 188, 179, 155, 143, 143,
     179, 179, 188
20095 DATA 188, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 129, 150, 129, 128, 128, 128, 128,
     129, 128, 128, 128, 128, 128, 128, 128, 128
20100 DATA 188, 182, 179, 147, 131, 185, 156, 128,
     128, 128, 136, 188, 140, 128, 128, 130, 179,
     179, 155,143, 128, 191, 191, 191, 191, 143, 143,
     143, 133, 170, 191, 143, 179, 188, 191, 191,
     191, 170
20105 DATA 149, 191, 163, 179, 191, 143, 143, 191,
     191, 191, 128, 149, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 129, 128, 128
20110 DATA 191, 191, 191, 191, 191, 188, 143, 128,
     128, 128, 128, 133, 128, 128, 128, 128, 138,
     143, 143, 179, 176, 179, 179, 140, 140, 179,
     179, 131, 131, 188, 188, 191, 191, 191, 191,
     191, 191, 178
20115 DATA 177, 191, 140, 132, 191, 162, 145, 191,
     191, 191, 128, 177, 128, 128, 128, 129, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
20120 DATA 191, 191, 191, 191, 191, 191, 182, 143,
     140, 172, 191, 191, 188, 176, 128, 128, 184,
     191, 191, 191, 143, 179, 188, 191, 143, 131,
     188, 191, 191, 188, 188, 188, 179, 179, 143,
     143, 191, 191
20125 DATA 191, 191, 191, 191, 191, 190, 189, 191,
     191, 191, 128, 191, 191, 180, 144, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
20130 DATA 191, 191, 191, 191, 191, 191, 191, 191,
     191, 189, 188, 188, 182, 179, 179, 179, 179,
     179, 185, 188, 191, 191, 191, 176, 143, 143,
     143, 191, 191, 191, 191, 191, 143, 135, 179,
     188, 188, 188
20135 DATA 179, 179, 143, 143, 191, 191, 191, 191,
     191, 191, 128, 191, 191, 191, 191, 180, 144,
     128, 128, 128, 128, 128, 128, 128, 128, 128
20140 DATA 191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 188, 188, 188, 179, 176, 179, 179, 143,
     143, 143, 191
20145 DATA 191, 190, 191, 191, 188, 188, 179, 179,
     143, 143, 176, 191, 191, 191, 191, 191, 191,
     180, 128, 128, 128, 129, 128, 128, 128, 128
20150 DATA 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 188
20155 DATA 188, 188, 179, 179, 179, 179, 143, 179,
     188, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 189, 180, 128, 128, 128, 128, 128, 128
20160 DATA 191,191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191
20165 DATA 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 189, 144, 128, 128, 128, 128
```




```
30000 REM ***** TOUCHDOWN GRAPHICS *****
30010 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
30015 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
30020 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
30025 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128,
     128,128,128,128,128,128,128,128,128,128
30030 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
30035 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
30040 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
30045 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
30050 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
30055 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
30060 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
```

```
30065 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
30070 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
30075 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 160, 152, 140, 140, 140, 140, 164,
     128, 128, 128, 128, 128, 128, 128, 128, 128
30080 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
30085 DATA 128, 128, 128, 128, 128, 128, 128, 160,
     152, 166, 185, 190, 191, 167, 179, 143, 185,
     129, 128, 128, 128, 128, 128, 128, 128, 128
30090 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 176, 176, 176, 176, 176, 176,
     176, 176, 176, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128
30095 DATA 128, 128, 128, 128, 160, 152, 166, 185,
     190, 191, 143, 179, 188, 191, 167, 134, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128
30100 DATA 128, 128, 128, 128, 128, 128, 128, 160,
     140, 140, 179, 140, 160, 188, 140, 176, 188,
     188, 188, 188, 179, 179, 179, 179, 179, 140,
     140, 140, 140, 140, 140, 176, 176, 176, 176,
     176, 176, 128
30105 DATA 128, 176, 152, 166, 185, 190, 191, 191,
     159, 137, 175, 191, 159, 185, 129, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128, 128
30110 DATA 128, 128, 128, 128, 128, 152, 134, 129,
     143, 143, 179, 147, 191, 128, 143, 143, 143,
     143, 191, 191, 191, 191, 191, 191, 191, 191,
     191, 191, 191, 191, 191, 188, 188, 188, 188,
     188, 188, 131
30115 DATA 179, 176, 178, 179, 147, 163, 179, 179,
     131, 131, 129, 128, 136, 172, 137, 140, 144,
     128, 128, 128, 128, 128, 128, 128, 128, 128
30120 DATA 128, 128, 128, 128, 128, 137, 144, 128,
     128, 128, 141, 133, 143, 143, 143, 191, 191,
     189, 188, 188, 188, 188, 188, 180, 179, 179,
     179, 179, 179, 179, 147, 143, 143, 143, 143,
     135, 180, 139
30125 DATA 191, 191, 191, 191, 189, 150, 188, 180,
     139, 143, 132, 128, 128, 160, 191, 191, 170,
     128, 128, 128, 128, 128, 128, 128, 128, 128
30130 DATA 128, 128, 128, 128, 128, 128, 130, 131,
     172, 133, 175, 140, 140, 140, 140, 140, 176,
     176, 176, 176, 176, 128, 128, 128, 131, 179,
     179, 179, 179, 143, 143, 143, 191, 191, 191,
     191, 191, 191
30135 DATA 188, 188, 188, 148, 140, 176, 179, 179,
     132, 128, 130, 171, 132, 131, 131, 136, 141,
     140, 140, 140, 140, 176, 176, 128, 128, 128
30140 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 131, 129, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 131, 131, 171, 130, 131,
     143, 143, 143, 143, 143, 143, 140, 188, 188,
     179, 179, 143
30145 DATA 131, 179, 179, 177, 155, 188, 188, 188,
     177, 176, 176, 180, 172, 188, 188, 146, 143,
     143, 143, 143, 143, 140, 140, 153, 128, 128
30150 DATA 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 128, 128, 128,
     128, 128, 128, 128, 128, 128, 130, 140, 140,
     140, 176, 136, 133, 143, 147, 131, 140, 140,
     140, 140, 172
30155 DATA 140, 132, 140, 156, 140, 140, 140, 140,
     140, 140, 140, 140, 140, 140, 140, 140, 134, 131,
     131, 131, 131, 131, 131, 131, 128, 128, 128
30160 DATA 176, 176, 176, 176, 176, 176, 176, 176,
     176, 176, 176, 176, 176, 176, 176, 176, 176,
     176, 176, 176, 176, 176, 176, 176, 176, 176,
     176, 178, 188, 188, 188, 177, 176, 176, 176,
     176, 176, 178
30165 DATA 188, 188, 188, 177, 176, 176, 176, 176,
     176, 176, 176, 176, 176, 176, 176, 176, 176,
     176, 176, 176, 176, 176, 176, 176, 176, 176
39999 REM ***** SCREEN-PRINT ROUTINE *****
40000 PRINT@55, "<P>RINT";
     : IF INKEY$ = "P" THEN GOTO 40010 ELSE RETURN
40010 FOR X = 0 TO 127
     : RESET(X,0)
     : RESET(X,47)
     : NEXT X
40020 FOR Y = 0 TO 47
     : RESET(0,Y)
     : RESET(127,Y)
     : NEXT Y
40030 POKE 16424,66
     : POKE 16425,1
40040 FOR Y = 0 TO 47
40050 LPRINT CHR$(30);TAB(10);
40060 FOR X = 0 TO 127
40070 IF POINT(X,Y) = -1 THEN LPRINT
     CHR$(18);CHR$(28);CHR$(3
40080 NEXT X
     : LPRINT CHR$(13);
     : NEXT Y
40090 LPRINT CHR$(30);
     : LPRINT CHR$(11)
40100 RETURN
50000 Z$ = INKEY$
     IF Z$ = "" THEN PRINT@1022, ">";
     : FOR T = 1 TO 50
     : NEXT T
     : PRINT@1022, " ";
     : FOR T = 1 TO 50
     : NEXT T
     : GOTO 50000 ELSE RETURN
```

Computer Clubs

NUTMEG POCKET COMPUTER GUILD
c/o Len Smith
3 Barn Hill Rd.
Bloomfield, CT 06002
1-203-242-9515

QUAD CITY COCO CLUB
c/o John Greve
4211 Seventh Ave.
Rock Island, IL 61201
1-309-786-8187

TUG-DC/TRS-80 USER'S GROUP OF DELAWARE COUNTY
c/o Gary Diillio
1109 Madison Ave.
Prospect Park, PA 19076

TUSCARAWAS COUNTY PERSONAL COMPUTER CLUB
c/o Robert D. Hobart
231 Fifth Drive NW
New Philadelphia, OH 44663

VECTECH COMPCLUB
c/o Sal Warmann
15362 Fairlane Dr.
Livonia, MI 48154

Hydrate

James W. Wood
424 N. Missouri
Atwood, IL 61913

Chemistry teachers will appreciate this program. "Hydrate" allows your student to perform a % composition lab on a high resolution screen. The student will measure the mass of an empty crucible, a crucible with hydrate, and the crucible with the dehydrated chemical. Figure 1 is a screen of

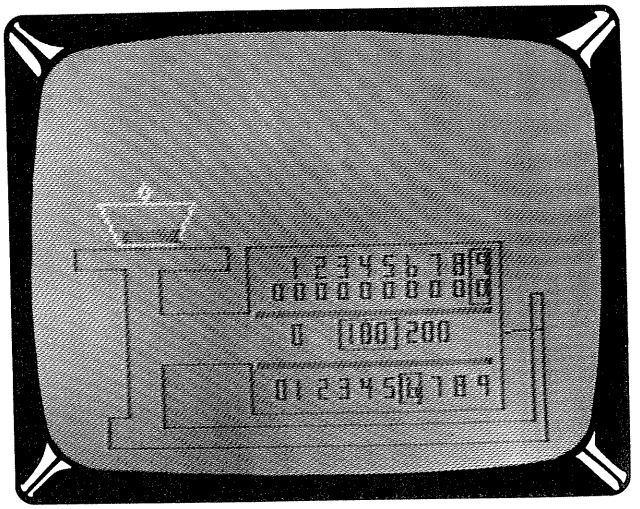


Figure 1. Triple beam balance.

the triple beam balance used to measure mass. Each run of the program will provide a different answer, thus the student cannot memorize the answer or pass it along to another student. This lab should not be used to replace a chemical lab, but to supplement the actual lab experience. This program is written for the Extended Color BASIC computer.

```
1 'JAMES W. WOOD,424 N. MISSOURI,ATWOOD, IL, 61913
2 W$(Ø)="D9R4U9L4"
      : W$(1) = "D9"
      : W$(2) = "R4D4L4D5R4"
      : W$(3) = "R4D4L4R4D5L4"
      : W$(4) = "D4R4U4D9"
      : W$(5) = "R4L4D4R4D5L4"
      : W$(6) = "D9R4U4L4"
      : W$(7) = "R4D9"
      : W$(8) = "R4D4L4U4D9R4U5"
      : W$(9) = "D4R4U4L4R4D9"
3 M(1) = 1ØØ+RND(25)
      : M(2) = 2Ø+RND(15)
      : M(3) = 4Ø+RND(15)
      : CLS
      : PRINT "% COMPOSITION LAB"
      : PRINT "FIND % WATER IN HYDRATE"
      : INPUT "NAME";NA$
4 CLS
      : PRINT
      : PRINT "INSTRUCTIONS FOR BALANCE"
      : PRINT
      : PRINT"TO MOVE----------USE"
5 PRINT
      : PRINT"1Ø GRAM          Q,W"
6 PRINT "1ØØ GRAM          A,S"
7 PRINT "1 GRAM            Z,X"
8 PRINT "                 <   >"
9 PRINT "PRESS 'ENTER' TO CONTINUE"
      : PRINT "AFTER USING SCALES."
1Ø PRINT
      : PRINT
      : PRINT "PRESS ANY KEY TO CONTINUE"
11 A$ = INKEY$
      : IF A$ = ""THEN 11
2Ø CLS
      : PRINT "FIRST FIND THE MASS OF CRUCIBLE."
      : PRINT
      : PRINT "YOU WILL WANT TO RECORD ALL"
      : PRINT "MEASUREMENTS AS THEY ARE MADE."
      : PRINT
21 PRINT
      : PRINT "PRESS ANY KEY TO CONTINUE"
      : M = M(1)
22 A$ = INKEY$
      : IFA$ = "" THEN22
25 GOSUB 2ØØØ
      : GOSUB1ØØØ
1ØØ CLS
      : PRINT NA$
      : PRINT
      : PRINT "NOW MEASURE MASS OF CRUCIBLE"
      : PRINT "AND THE HYDRATE."
      : PRINT
      : PRINT "PRESS ANY KEY TO CONTINUE"
1Ø1 M = M(1)+M(2)+M(3)
1Ø2 A$ = INKEY$
      : IF A$ = ""THEN 1Ø2
1Ø5 GOSUB 2ØØØ
      : GOSUB 3ØØØ
      : GOSUB 1ØØØ
2ØØ CLS
      : PRINT NA$
      : PRINT "TIME TO HEAT THE CHEMICAL."
      : PRINT "YOU MUST WAIT FOR THE CHEMICAL"
      : PRINT "TO HEAT AND COOL PROPERLY."
      : PRINT "WATCH THE CHEMICAL TURN COLOR"
      : PRINT "AS THE HYDRATED WATER ESCAPES."
2Ø1 PRINT
      : PRINT "PRESS ANY KEY TO CONTINUE"
2Ø2 A$ = INKEY$
      : IFA$ = "" THEN 2Ø2
2Ø5 GOSUB 2ØØØ
      : GOSUB 3ØØØ
21Ø COLOR 4,1
      : LINE(4Ø,11Ø)-(5Ø,18Ø),PSET,BF
212 LINE(1Ø,18Ø)-(8Ø,185),PSET,BF
22Ø COLOR 3,1
      : LINE(4Ø,11Ø)-(45,92),PSET
      : LINE-(5Ø,11Ø),PSET
      : LINE-(4Ø,11Ø),PSET
      : PAINT(45,99),3,3
23Ø FOR CC = 1 TO 1ØØØ
      : PSET(3Ø+RND(31),81+RND(6),2)
      : NEXT CC
235 COLOR 1,1
      : LINE(4Ø,9Ø)-(5Ø,11Ø),PSET,BF
      : FOR TI = 1 TO 15ØØ
      : NEXT TI
3ØØ CLS
      : PRINT NA$
      : PRINT
      : PRINT "FIND MASS OF CRUCIBLE AND THE"
      : PRINT "DEHYDRATED CHEMICAL."
      : PRINT
      : PRINT "PRESS ANY KEY TO CONTINUE"
3Ø1 A$ = INKEY$
      : IF A$ = ""THEN 3Ø1
3Ø2 GOSUB 2ØØØ
      : COLOR 2,1
      : LINE(32,82)-(61,87,,PSET,BF
      : M = M(1)+M(3)
      : GOSUB 1ØØØ
4ØØ PRINT NA$
      : PRINT
      : PRINT "NOW TO CALCULATE'
```

```
41Ø PRINT
     : INPUT "MASS OF EM TY CRUCIBLE";N(1)
     : IF N(1) <> M(1) THEN PRINT "FUNNY, I GOT";M(1)
     ELSE PRINT "CORRECT"
42Ø INPUT "MASS OF CRUCIBLE AND HYDRATE";N(2)
     : IF N(2) <> M(1)+M(2)+M(3) THEN PRINT "STRANGE,
     IT SHOULD BE";M(1)+M(2)+M(3) ELSE PRINT
     "CORRECT"
43Ø PRINT "MASS OF CRUCIBLE AND CHEMICAL"
     : INPUT "AFTER HEATING";N(3)
     : IF N(3) <> M(1)+M(3) THEN PRINT "TOO BAD,
     ITS";M(1)+M(3) ELSE PRINT "CORRECT"
44Ø PRINT "THE % WATER WOULD BE EQUAL TO"
     : PRINT "THE MASS OF WATER DIVIDED BY"
     : PRINT "MASS OF THE ORIGINAL CHEMICAL"
     : PRINT "TIMES 1ØØ."
45Ø INPUT "WHAT IS THE MASS OF WATER";MW
46Ø IF MW = M(2) THEN PRINT "CORRECT"
     : GOTO 475
47Ø PRINT "TRY MASS DISH WITH ORIGINAL"
     : PRINT "CHEMICAL MINUS MASS DISH WITH"
     : PRINT "DEHYDRATED CHEMICAL."
     : INPUT "TRY AGAIN, MASS WATER";MW
     : IF MW <> M(2) THEN PRINT "IT IS";M(2) ELSE
     PRINT "NOW YOU'RE CORRECT ";NA$
475 INPUT "MASS ORIGINAL CHEMICAL";OC
48Ø IF OC = M(2)+M(3) THEN PRINT "CORRECT"
     : GOTO 5ØØ
49Ø PRINT "TRY SUBTRACTING MASS OF EMPTY"
     : PRINT "CRUCIBLE FROM MASS OF CRUCIBLE"
     : PRINT "WITH ORIGINAL CHEMICAL."
     : INPUT "AGAIN, ORIGINAL CHEMICAL";OC
     : IF OC <> M(2)+M(3) THEN PRINT "IT
     IS";M(2)+M(3) ELSE PRINT "NOW YOU'RE CORRECT
     ";NA$
5ØØ INPUT "WHAT IS THE % WATER";PW
51Ø IF ABS(PW-1ØØ*M(2)/(M(2)+M(3))) > 1 THEN PRINT
     "TRY ";M(2);"*1ØØ/";M(2)+M(3)
     : GOTO 515 ELSE PRINT "WE MADE IT! ";NA$
     : GOTO 52Ø
515 INPUT "% WATER";PW
516 IF ABS(PW-1ØØ*M(2)/(M(2)+M(3))) > 1 THEN PRINT
     "ACTUALLY IT IS";1ØØ*M(2)/(M(2)+M(3));"%" ELSE
     PRINT "NOW YOU'RE CORRECT"
52Ø PRINT "ANOTHER LAB (Y/N)
53Ø A$ = INKEY$
     : IF A$ = "Y" THEN RUN ELSE IF A$ = "N" THEN END
     ELSE 53Ø
54Ø PRINT "GOODBYE ";NA$
9ØØ STOP
1ØØØ Q1 = Ø
     : A1 = Ø
     : Z1 = Ø
     : COLOR 4,1
1Ø1Ø DRAW"BM6,9ØR8ØD1ØL35D2ØR45U3ØR134D42R5L5D36L134U
     24L45D3ØR195U6ØR7D7ØL23ØU15R1ØU7ØL27U8"
1Ø2Ø LINE(246,131)-(25Ø,131),PSET
1Ø3Ø LINE(1ØØ,12Ø)-(225,122),PSET,BF
1Ø4Ø LINE(1ØØ,144)-(225,146),PSET,BF
1Ø5Ø DRAW"BM11Ø,152XW$(Ø);"
1Ø6Ø DRAW"BM122,152XW$(1);"
1Ø7Ø DRAW"BM134,152XW$(2);"
1Ø8Ø DRAW"BM146,152XW$(3);"
1Ø9Ø DRAW"BM158,152XW$(4);"
11ØØ DRAW"BM17Ø,152XW$(5);"
111Ø DRAW"BM182,152XW$(6);"
112Ø DRAW"BM194,152XW$(7);"
113Ø DRAW"BM2Ø6,152XW$(8);"
114Ø DRAW"BM218,152XW$(9);"
115Ø DRAW"BM12Ø,126XW$(Ø);"
116Ø DRAW"BM152,126XW$(1);"
117Ø DRAW"BM158,126XW$(Ø);"
118Ø DRAW"BM166,126XW$(Ø);"
119Ø DRAW"BM182,126XW$(2);"
12ØØ DRAW"BM19Ø,126XW$(Ø);"
121Ø DRAW"BM198,126XW$(Ø);"
122Ø FORX = 11ØTO218STEP12
123Ø LINE(X,1Ø7)-(X+4,114),PSET,B
     : NEXTX
124Ø DRAW"BM122,95XW$(1);"
125Ø DRAW"BM134,95XW$(2);"
126Ø DRAW"BM146,95XW$(3);"
127Ø DRAW"BM158,95XW$(4);"
128Ø DRAW"BM17Ø,95XW$(5);"
129Ø DRAW"BM182,95XW$(6);"
13ØØ DRAW"BM194,95XW$(7);"
131Ø DRAW"BM2Ø6,95XW$(8);"
132Ø DRAW"BM218,95XW$(9);"
133Ø B1 = 1Ø7
     : B2 = 117
     : B3 = 93
     : B4 = 124
     : B5 = 15Ø
     : C1 = 1Ø7
134Ø LINE(B1,B3)-(B1+9,B3+25),PSET,B
135Ø LINE(B2,B4)-(B2+3Ø,B4+14),PSET,B
136Ø LINE(B1,B5)-(B1+9,B5+14),PSET,B
137Ø LINE(235,131)-(245,122),PSET
138Ø A$ = INKEY$
     : IF A$ = ""THEN 138Ø
139Ø IF ASC(A$) = 13 THEN RETURN
14ØØ IF A$ = "Q"THEN Q1 = Q1-1
     : IF Q1 < Ø THEN Q1 = Ø ELSE
     LINE(B1,B3)-(B1+9,B3+25),PRESET,B
     : B1 = B1-12
     : LINE(B1,B3)-(B1+9,B3+25),PSET,B
     : GOTO 147Ø
141Ø IF A$ = "W" THEN Q1 = Q1+1
     : IF Q1 > 9 THEN Q1 = 9 ELSE
     LINE(B1,B3)-(B1+9,B3+25),PRESET,B
     : B1 = B1+12
     : LINE(B1,B3)-(B1+9,B3+25),PSET,B
     : GOTO 147Ø
142Ø IF A$ = "A" THEN A1 = A1-1
     : IF A1 < Ø THEN A1 = Ø ELSE
     LINE(B2,B4)-(B2+3Ø,B4+14),PRESET,B
     : B2 = B2-3Ø
     : LINE(B2,B4)-(B2+3Ø,B4+14),PSET,B
     : GOTO 147Ø
143Ø IF A$ = "S" THEN A1 = A1+1
     : IF A1 > 2 THEN A1 = 2 ELSE
     LINE(B2,B4)-(B2+3Ø,B4+14),PRESET,B
     : B2 = B2+3Ø
     : LINE(B2,B4)-(B2+3Ø,B4+14),PSET,B
     : GOTO 147Ø
144Ø IF A$ = "Z" THENZ1 = Z1-1
     : IF Z1 < Ø THEN Z1 = Ø ELSE
     LINE(C1,B5)-(C1+9,B5+14),PRESET,B
     : C1 = C1-12
     : LINE(C1,B5)-(C1+9,B5+14),PSET,B
     : GOTO 147Ø
145Ø IF A$ = "X" THEN Z1 = Z1+1
     : IF Z1 > 9 THEN Z1 = 9 ELSE
     LINE(C1,B5)-(C1+9,B5+14),PRESET,B
     : C1 = C1+12
     : LINE(C1,B5)-(C1+9,B5+14),PSET,B
     : GOTO 147Ø
146Ø GOTO 138Ø
147Ø IF M > Q1*1Ø+A1*1ØØ+Z1 THEN
     LINE(235,131)-(245,122),PSET
     : LINE(235,131)-(245,131),PRESET
     : LINE(235,131)-(245,14Ø),PRESET
     : GOTO 138Ø
148Ø IF Q1*1Ø+A1*1ØØ+Z1 = M THEN
     LINE(235,131)-(245,122),PRESET
     : LINE(235,131)-(245,131),PSET
     : LINE(235,131)-(245,14Ø),PRESET
     : GOTO 138Ø
149Ø IF Q1*1Ø+A1*1ØØ+Z1>M THEN
     LINE(235,131)-(245,122),PRESET
     : LINE(235,131)-(245,131),PRESET
     : LINE(235,131)-(245,14Ø),PSET
     : GOTO 138Ø
```

```
2000 PMODE 3,1
     : PCLS
     : SCREEN 1,0
2010 COLOR 2,1
     : LINE(20,70)-(30,88),PSET
     : LINE-(60,88),PSET
     : LINE-(70,70),PSET
     : LINE-(20,70),PSET
     : CIRCLE(45,65),5
2050 RETURN
3000 COLOR 3,1
     : LINE(32,82)-(61,87),PSET,BF
3050 RETURN
```

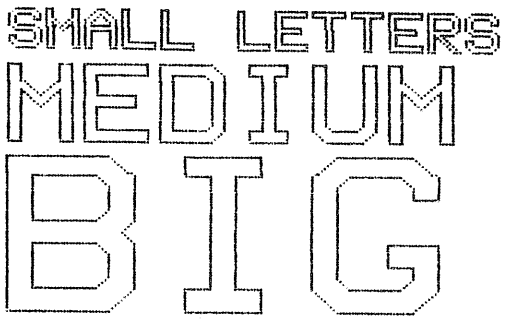
Lettering on the Color Computer

Ronald T. Graff
212 Sandy Lane Apt 104D
Warwick, RI 02889

This program draws nice neat letters in small (up to 21 characters per line), medium (up to 10 characters per line), and large (up to 5 characters per line) sizes on the Extended BASIC Color Computer.

```
0 CLS
     : DIM A$(95)
     : GOSUB 1000
1 CLS
     : PRINT "DO YOU WANT (1) SMALL (2) MEDIUM (3)
     LARGE"
     : INPUT CH
2 IF CH = 1 THEN Y = -16
     : YY = 16
     : XX = 12
     : S = 1 ELSE IF CH = 2 THEN Y = -32
     : YY = 32
     : XX = 24
     : S = 2 ELSE IF CH = 3 THEN Y = -64
     : YY = 64
     : XX = 48
     : S = 4 ELSE 1
3 IF CH = 1 THEN CLS
     : PRINT "ENTER NEXT LINE (21 CHARACTERS)" ELSE
     IF CH = 2 THEN CLS
     : PRINT "ENTER NEXT LINE (10 CHARACTERS)" ELSE
     IF CH = 3 THEN CLS
     : PRINT "ENTER NEXT LINE (5 CHARACTERS)"
4 A$ = ""
     : LINE INPUT A$
     : CLS
5 IF A$ = "" GOTO 3
10 PMODE 4,1
     : SCREEN 1,0
20 COLOR 0,1
     : IF Y = -YY THEN PCLS
30 Y = Y+YY
     : P = 0
     : FOR X = 0 TO (LEN(A$)-1)*XX STEP XX
40 XY$ = "S"+STR$(S)+"BM"+STR$(X)+","+STR$(Y)
50 P = P+1
     : N = ASC(MID$(A$,P,1))
60 DRAW XY$+A$(N)
70 NEXT
80 A$ = INKEY$
     : IF A$ = CHR$(13) THEN 3 ELSE IF A$ = CHR$(12)
     THEN 1 ELSE 80
1000 A$(33) = "BR16R8D40L8U40BD48R8D8L8U8"
1010 A$(34) = "BR8R8D24L8U24BR16R8D24L8U24"
1020 A$(35) = "BR8R8D16R8U16R8D16R8D8L8D8R8D8L8
     D16L8U16L8D16L8U16L8U8R8U8L8U8R8U16BD24BR8
     R8D8L8U8"
1030 A$(36) = "BR16R8D8R16D8L16D8R8F8D8G8L8D8L8
     U8L16U8R16U8L8H8U8E8R8U8BD16D8H4E4BD16BR8
     F4G4U8"
1040 A$(37) = "R16D16L16U16BD4BR40G40D8E40U8BD36
     BL16R16D16L16U16"
1050 A$(38) = "BR8R8F8D16G4F8E4R8D8G4F4D8L8H4G4
     L16H8U16E4H4U16E8BD8BR4F4D8G4H4U8E4BD24F12G4
     L8H4U8E4"
1060 A$(39) = "BR16R8D24L8U24"
1070 A$(40) = "BR16R8G16D24F16L8H16U24E16"
1080 A$(41) = "BR16R8F16D24G16L8E16U24H16"
1090 A$(42) = "BR16R8D16E8R8D8G12F12D8L8H8D16L8
     U16G8L8U8E12H12U8R8F8U16"
1100 A$(43) = "BD24R16U16R8D16R16D8L16D16L8U16L16U8"
1110 A$(44) = "BD32BR16R8D16G8L8E8U16"
1120 A$(45) = "BD24R40D8L40U8"
1130 A$(46) = "BD48BR16R8D8L8U8"
1140 A$(47) = "BD4BR40D8G40U8E40"
1150 A$(48) = "BR8R24F8D40G8L24H8U40E8BD8BR4R16
     F4G24U24E4BD12BR20D24G4L16H4E24"
1160 A$(49) = "BR16R8D48R8D8L24U8R8U32L8U8E8"
1170 A$(50) = "BR8R24F8D8G32R32D8L40U8E32U4H4L16
     G4D4L8U8E8"
1180 A$(51) = "BR8R24F8D16G4F4D16G8L24H8U8R8D4F4
     R16E4U8H4L8U8R8E4U8H4L16G4D4L8U8E8"
1190 A$(52) = "BR24R8D32R8D8L8D16L8U16L24U16E24
     BD8D24L16U8E16"
1200 A$(53) = "R40D8L32D8R24F8D24G8L24H8U8R8D4F4
     R16E4U16H4L28U24"
1210 A$(54) = "BR8R24F8D8L8U4H4L16G4D12R24F8D16
     G8L24H8U40E8BD32R20F4D8G4L16H4U12"
1220 A$(55) = "R40D8G32D16L8U16E32L32U8"
1230 A$(56) = "BR8R24F8D16G4F4D16G8L24H8U16E4H4
     U16E8BD8BR4R16F4D8G4L16H4U8E4BD24R16F4D8G4
     L16H4U8E4"
1240 A$(57) = "BR8R24F8D40G8L24H8U8R8D4F4R16E4
     U12L24H8U16E8BD8BR4R16F4D12L20H4U8E4"
1250 A$(58) = "BD16BR16R8D8L8U8BD16R8D8L8U8"
1260 A$(59) = "BD16BR16R8D8L8U8BD16R8D16G8L8E8U16"
1270 A$(60) = "BR34D8G20F20D8H28E28"
1280 A$(61) = "BD16R40D8L40U8BD16R40D8L40U8"
1290 A$(62) = "BR6F28G28U8E20H20U8"
1300 A$(63) = "BR8R24F8D8G16D8L8U8E16U4H4L16G4D4
     L8U8E8BD48BR8R8D8L8U8"
1310 A$(64) = "BR8R24F8D24G8L16U24R8D8R8U12H4L16
     G4D32F4R28D8L32H8U40E8"
1320 A$(65) = "BR16R8F16D40L8U24L24D24L8U40E16BD8
     BR4F12D4L24U4E12"
1330 A$(66) = "R32F8D16G4F4D16G8L32U56BF8R20F4D8
     G4L20U16BD24R20F4D8G4L20U16"
1340 A$(67) = "BR8R24F8D8L8U4H4L16G4D32F4R16E4U4
     R8D8G8L24H8U40E8"
1350 A$(68) = "R32F8D40G8L32U56BF8R20F4D32G4L20U40"
1360 A$(69) = "R40D8L32D16R24D8L24D16R32D8L40U56"
1370 A$(70) = "R40D8L32D16R24D8L24D24L8U56"
1380 A$(71) = "BR8R24F8D8L8U4H4L16G4D32F4R16E4U4
     L8U8R16D16G8L24H8U40E8"
1390 A$(72) = "R8D24R24U24R8D56L8U24L24D24L8U56"
1400 A$(73) = "BR8R24D8L8D40R8D8L24U8R8U40L8U8"
1410 A$(74) = "BR32R8D48G8L24H8U8R8D4F4R16E4U44"
1420 A$(75) = "R8D24E24R8G28F28L8H24D24L8U56"
1430 A$(76) = "R8D48R32D8L40U56"
1440 A$(77) = "R8D12F12E12U12R8D56L8U36G12H12
     D36L8U56"
1450 A$(78) = "R8D12F24U36R8D56L8U12H24D36L8U56"
1460 A$(79) = "BR8R24F8D40G8L24H8U40E8BD8BR4R16
     F4D32G4L16H4U32E4"
1470 A$(80) = "R32F8D16G8L24D24L8U56BF8R20F4D8
     G4L20U16"
1480 A$(81) = "BR8R24F8D36G4F4G4H4G4L20H8U40E8B
     D8BR4R16F4D32H4G4F4L16H4U32E4"
1490 A$(82) = "R32F8D16G8L16F24L8H24D24L8U56BF8
     R20F4D8G4L20U16"
```

```
150Ø A$(83) = "BR8R24F8D8L8U4H4L16G4D8F4R2ØF8D16
     G8L24H8U8R8D4F4R16E4U8H4L2ØH8U16E8"
151Ø A$(84) = "R4ØD8L16D48L8U48L16U8"
152Ø A$(85) = "R8D44F4R16E4U44R8D48G8L24H8U48"
153Ø A$(86) = "R8D36F12E12U36R8D4ØG16L8H16U4Ø"
154Ø A$(87) = "R8D36E12F12U36R8D56L8U12H12G12
     D12L8U56"
155Ø A$(88) = "R8D12F12E12U12R8D16G12F12D16L8U12
     H12G12D12L8U16E12H12U16"
156Ø A$(89) = "R8D12F12E12U12R8D16G16D24L8U24H16U16"
157Ø A$(9Ø) = "R4ØD12G32D4R32D8L4ØU12E32U4L32U8"
158Ø A$(91) = "R4ØD8L24D4ØR24D8L4ØU56"
159Ø A$(92) = "BD4F4ØD8H4ØU8"
16ØØ A$(93) = "R4ØD56L4ØU8R24U4ØL24U8"
161Ø A$(94) = "BD34E2ØF2ØD8H2ØG2ØU8"
162Ø A$(95) = "BD48R4ØD8L4ØU8"
1999 RETURN
```



Graphic Printouts for the CoCo and the CGP-115

Howard C. Price
R.R. 2, Box 84
Warsaw, IN 46580

This program can be used to vary the printouts using the Color Computer and the CGP-115 Color Graphics printer.

```
1Ø CLS
15 PRINT"CHARACTERS PER LINE=8Ø/(SIZE+1)
2Ø INPUT"TYPE SIZE";N
25 PRINT"THE MAXIMUM NUMBER OF CHARACTERS AND SPACES
     YOU CAN TYPE PER LINE IS ......";INT(8Ø/(N+1))
3Ø PRINT#-2, CHR$(18)
4Ø PRINT#-2,"S";STR$(N)
5Ø LINE INPUT"TYPE OR REPEAT(#)        ";A$
51 IF A$= "#" THEN 74 ELSE 6Ø
6Ø PRINT#-2,"P";A$
64 PRINT#-2,CHR$(17)
66 FOR T = 1 TO 292
     : NEXT T
67 C$ = INKEY$
68 IF C$ = "^" THEN GOSUB 9Ø ELSE 71
71 PRINT#-2,CHR$(18)
72 LINE INPUT"TYPE OR REPEAT(#)        ";B$
73 IF B$ = "#" THEN 6Ø ELSE74
74 PRINT#-2,"P";B$
75 PRINT#-2,CHR$(17)
76 FOR T = 1 TO 292
     : NEXT T
77 C$ = INKEY$
78 IF C$ = "^" THEN GOSUB 9Ø ELSE 3Ø
8Ø GOTO 3Ø
9Ø PRINT#-2,CHR$(18)
95 PRINT"LINE FEED IS";(N+1)*12
1ØØ INPUT"HOW MUCH ROLL BACK(MAX=9ØØ)";M
11Ø PRINT#-2,"RØ,";STR$(M)
12Ø INPUT"TYPE OR ROLL BACK, T OR R";F$
13Ø IF F$ = "R" THEN 1ØØ ELSE RETURN
```

SINE/COSINE/TANGENT Graph

Charles Robert Kelly II
P.O. Box 1862
Rolla, MO 65401

This program is one I have written over the past several months. The program will draw any of the sine/cosine/tangent graphs.

The program is also capable of producing a hard copy on a line-printer. For example, the use of the Line-Printer VII with the TRS-80 Extended BASIC Color Computer requires the use of the Screen Print Program available from Radio Shack. In addition the program requires a 16K memory as it uses almost 5K.

Instructions for the operation of the program are included in the program.

```
1Ø CLEAR 5ØØ
     : P = Ø
     : B$ = ""
     : A$ = ""
     : C$ = ""
     : FUNC = Ø
2Ø CLS
     : DATA 195,"SINE/COSINE/TANGENT GRAPH BY CHARLES
     ROBERT KELLY II  MAY 1Ø, 1982"4Ø DATA 2Ø3,"
     CAUTION THIS PROGRAM REQUIRES THE USE OF THE
     SCREEN PRINT PROGRAM"
5Ø CLS
     : FOR WW = 1 TO 2
     : READ QQ,K$
     : FOR J = 1 TO LEN(K$)
51 PRINT@ QQ,LEFT$(K$,J)
52 FOR KK = 1 TO 3
     : NEXT
53 NEXT
     : FOR SS = 1 TO 15ØØ
     : NEXT
     : CLS
     : NEXT
6Ø GOSUB 99Ø
7Ø CLS
     : PRINT"WILL YOU WANT A HARD COPY OF ANY OF THE
     GRAPHS YOU WILL BE CREATING? (Y/N)"
8Ø L$ = INKEY$
9Ø IF L$ = "Y" THEN GOTO 11ØØ
1ØØ IF L$ = "N" THEN A$ = "N"
11Ø IF L$ = "N" THEN GOTO17Ø
12Ø IF L$ = "" THEN GOTO 8Ø
13Ø P = Ø
     : B$ = ""
     : A$ = ""
     : C$ = ""
     : FUNC = Ø
14Ø DEFUSRØ = 15785
15Ø P = Ø
     : PRINT"WILL YOU WANT A PRINT OF THIS GRAPH (Y
     OR N)?"
     : C$ = ""
16Ø A$ = INKEY$
17Ø IF A$ = "Y" THEN P = 1
18Ø IF A$ = "N" THEN P = 2
19Ø IF P = Ø THEN GOTO 16Ø
2ØØ FOR X = 1 TO 1ØØ
     : NEXT X
21Ø CLS
     : PRINT"PLEASE ENTER THE NAME OF THE FUNCTION
     YOU WISH TO VIEW. (SINE,COSINE,TANGENT)"
     : C$ = ""
```

```
22Ø B$ = INKEY$
23Ø IF B$ = "S" THEN C$ = "SINE"
24Ø IF B$ = "C" THEN C$ = "COSINE"
25Ø IF B$ = "T" THEN C$ = "TANGENT"
26Ø IF C$ = "" THEN 22Ø
27Ø IF C$ = "SINE" THEN FUNC = 1
28Ø IF C$ = "COSINE" THEN FUNC = 2
29Ø IF C$ = "TANGENT" THEN FUNC = 3
3ØØ CLS
     : INPUT "ENTER THE DENSITY YOU WANT THE GRAPH
     PRINTED. THE LOWER THE NUMBER, THE GREATER THE
     DENSITY. (HIT THE ENTER KEY)";DEN
31Ø CLS
     : INPUT"ENTER THE AMPLITUDE YOU WANT FOR THE
     GRAPH (-3 TO 3)";AMP
32Ø IF AMP<-3 THEN GOTO 31Ø
33Ø IF AMP> 3 THEN GOTO 31Ø
34Ø CLS
     : INPUT"ENTER THE FREQUENCY THAT YOU WANT ON THE
     GRAPH. HIT THE ENTER KEY.";FREQ
35Ø CLS
     : PRINT@192,"THE SCREEN WHICH IS SHOWN HAS THE X
     AND Y SCALES DIVIDED INTO 1/2 UNITS."
36Ø FOR X = 1 TO 1ØØØ
     : NEXT X
37Ø PMODE 4,1
38Ø PCLS
39Ø SCREEN 1,1
4ØØ GOSUB 87Ø
41Ø GOSUB 93Ø
42Ø LINE (127,5)-(127,185),PSET
43Ø LINE(7,95)-(247,95),PSET
44Ø GOSUB 84Ø
45Ø IF FREQ = Ø THEN FREQ = 1
46Ø IF AMP = Ø THEN AMP = 1
47Ø IF DEN = Ø THEN DEN = 1
48Ø FOR X = -18Ø TO 18Ø STEP DEN
49Ø AX = (X/57.29578)*FREQ*6
5ØØ XP = X/1.5+127
51Ø ON FUNC GOTO 54Ø,52Ø,56Ø
52Ø F1 = -(COS(AX)*3Ø*AMP)+95
53Ø GOTO 59Ø
54Ø F1 = -(SIN(AX)*3Ø*AMP)+95
55Ø GOTO 59Ø
56Ø F1 = -(TAN(AX)*3Ø*AMP)+95
57Ø IF F1 > 186 THEN 6ØØ
58Ø IF F1 < 5 THEN 6ØØ
59Ø PSET(XP,F1,3)
6ØØ NEXT X
61Ø IF P <> 1 THEN 73Ø
62Ø PRINT #-2,CHR$(26)
63Ø PRINT #-2,C$;" GRAPH";
64Ø PRINT #-2,"  FREQUENCY ";FREQ;
65Ø PRINT #-2,"  AMPLITUDE ";AMP;
66Ø PRINT #-2,"  DENSITY ";DEN
67Ø FOR X = 1 TO 3
     : PRINT #-2,CHR$(26)
     : NEXT X
68Ø POKE 163Ø3,Ø
69Ø Y = USR(Ø)
7ØØ FOR XX = 1 TO 1Ø
71Ø PRINT#-2,CHR$(26)
72Ø NEXT XX
73Ø GOSUB 1ØØØØ
735 K$ = INKEY$
74Ø IF K$ = ""THEN735
75Ø CLS
     : PRINT"DO YOU HAVE ANOTHER GRAPH TO DRAW?
     (Y/N)"
76Ø J$ = INKEY$
77Ø FOR X = 1TO 1ØØ
     : NEXT X
78Ø IF J$ = "Y" GOTO 79Ø ELSE 81Ø
79Ø IF L$ = "N" GOTO 21Ø
8ØØ IF J$ = "Y" GOTO 15Ø
81Ø IF J$ = "N" THEN GOTO 83Ø
82Ø GOTO 76Ø
83Ø CLS
     : END
84Ø LINE(Ø,Ø)-(254,19Ø),PSET,B
85Ø LINE(2,2)-(252,188),PSET,B
855 LINE(8,183)-(81,167),PSET,B
86Ø RETURN
87Ø FOR X = 7 TO 247 STEP 2Ø
88Ø FOR Y = 92 TO 98
89Ø PSET(X,Y)
9ØØ NEXT Y
91Ø NEXT X
92Ø RETURN
93Ø FOR Y = 5 TO 195 STEP 15
94Ø FOR X = 124 TO 13Ø
95Ø PSET(X,Y)
96Ø NEXT X
97Ø NEXT Y
98Ø RETURN
99Ø DATA"THIS PROGRAM IS DESIGNED TO DRAW A GRAPH OF
     YOUR DESIGN ON THE VIDEO SCREEN. IN ADDITION IF
     YOU WISH, THE PROGRAM WILL COPY THE GRAPH ONTO A
     LINE PRINTER. THIS PARTICULAR PROGRAM WAS
     DESIGNED FOR THE LINE PRINTER VII"
991 CLS
     : READ K$
     : FOR J = 1 TO LEN(K$)
992 PRINT@Ø, LEFT$(K$,J)
993 NEXT
1ØØØ GOSUB 2ØØØ
1Ø2Ø DATA Ø,"YOU WILL BE ASKED SEVERAL QUESTIONS TO
     SET THE PARAMETERS OF THE GRAPH. SOME OF THE
     QUESTIONS CAN BE ANSWERED BY THE USE OF ONE
     LETTER. IT WILL NOT BE NECESSARY TO HIT THE
     'ENTER' KEY AFTER THESE QUESTIONS."
1Ø3Ø DATA 224,"IF YOU DO NOT ENTER ANY VALUES FOR THE
     DENSITY, AMPLITUDE, OR FREQUENCY 1(ONE) IS
     USED."
1Ø31 CLS
     : FOR ZZ = 1 TO 2
     : READ QQ,K$
     : FOR J = 1 TO LEN(K$)
1Ø32 PRINT@QQ,LEFT$(K$,J)
1Ø33 NEXT
     : NEXT
1Ø5Ø GOSUB 2ØØØ
1Ø6Ø DATA "AFTER THE GRAPH HAS BEEN DRAWN AND AFTER
     IT HAS BEEN PRINTED, IF THAT WAS YOUR CHOICE,
     ANOTHER GRAPH MAY BE DRAWN. TO RESET THE
     SEQUENCE, TOUCH ANY KEY ON THE KEYBOARD EXCEPT
     THE 'BREAK' KEY."
1Ø81 CLS
     : READ K$
     : FOR J = 1 TO LEN(K$)
1Ø82 PRINT@Ø,LEFT$(K$,J)
1Ø83 NEXT
1Ø84 GOSUB 2ØØØ
1Ø9Ø RETURN
11ØØ F$ = ""
     : PRINT"HAVE YOU ALREADY ENTERED THE SCREEN
     PRINT PROGRAM? (Y/N)"
111Ø F$ = INKEY$
112Ø IF F$ = "Y" THEN GOTO15Ø
113Ø IF F$ = "" THEN GOTO111Ø
114Ø CLS
     : PRINT"PUT THE SCREEN PRINT PROGRAM INTO THE
     CASSETTE PLAYER AND PRESS THE PLAY KEY. HIT THE
     'ENTER' KEY WHEN YOU ARE READY.": INPUTW
115Ø CLEAR 2ØØ,15743
116Ø CLS
     : PRINT"THE SCREEN PRINT PROGRAM IS NOW
     LOADING....."
117Ø CLOADM
118Ø EXEC
119Ø CLS
     : PRINT"THE PROGRAM IS NOW LOADED. YOU MAY
     REWIND YOUR TAPE AND TAKE IT FROM THE CASSETTE
```

```
     PLAYER."
12ØØ GOTO 13Ø
2ØØØ PRINT@48Ø,"HIT 'ENTER' TO CONTINUE";
     : INPUTW
     : RETURN
1ØØØØ ' LABEL
1ØØ4Ø DRAW"C3;S4;BM-68,+12"
1ØØ41 IF C$ = "SINE" THEN AA$ = "SIN-GRAPH"
1ØØ42 IF C$ = "COSINE" THEN AA$ = "COS-GRAPH"
1ØØ45 IF C$ = "TANGENT" THEN AA$ = "TAN-GRAPH"
1ØØ5Ø GOSUB 13ØØØ
1ØØ6Ø RETURN
13ØØØ 'CHAR-GEN
13ØØ1 FOR XX = 1 TO 9
13ØØ2 RESTORE
     : LL = Ø
13ØØ3 READ LL$,CC$
13ØØ4 IF LL$ = MID$(AA$,XX,1) THEN DRAW CC$
     : GOTO 13ØØ6
13ØØ5 LL = LL+1
     : IF LL < 18 THEN13ØØ3
13ØØ6 NEXT
     : RETURN
13ØØ7 DATA" ","BM+7,Ø"
13Ø1Ø DATA"A","U4;E2;F2;D2;NL4;D2;BM+3,Ø"
13Ø11 DATA"C","BM+1,-Ø;H1;U4;E1;R2;F1;BM+Ø,4;G1;L2;
     BM+6,Ø"13Ø12 DATA"G","BM+1,-Ø;H1;U4;E1;R2;F1;
     BM+Ø,+2;NL1;D2;G1;L2;BM+6,Ø"
13Ø13 DATA"H","U3;NU3;R4;NU3;D3;BM+3,Ø"
13Ø14 DATA"I","BM+1,Ø;R1;NR1;U6;NL1;R1;BM+4,+6"
13Ø15 DATA"N","U6;F1;D1;F2;D1;F1;NU6;BM+3,Ø"
13Ø16 DATA"O","BM+1,Ø;H1;U4;E1;R2;F1;D4;G1;L2;BM+6,Ø"
13Ø17 DATA"P","U6;R3;F1;D1;G1;L3;BM+7,3"
13Ø18 DATA"S","BM+Ø,-1;F1;R2;E1;U1;H1;L2;H1;U1;E1;R2;
     F1;BM+3,+5"
13Ø19 DATA"T","BM+2,+Ø;U6;NL2;R2;BM+3,+6"
13Ø2Ø DATA"-","BM+Ø,-3;R4;BM+3,+3"
13Ø21 DATA"R","U6;R3;F1;D1;G1;L2;NL1;F3;BM+3,Ø"
```

Magic Square

Robert K. Phelps
143 Forest Park
7800 Tayloe Drive
Manassas, VA 22111

Magic Square is designed for use with the PC-2 with a printer/plotter attached. If the printer/plotter is not hooked up to the computer, the plotter commands will be ignored. Start the program by pressing the DEF SPACE keys.

The printing will normally be done using black and red, but the pens can be switched to get other color combinations. Also, you can change the COLOR commands in the program to get different color effects.

4 * 4 MAGIC SQUARE STARTING WITH 100

100	114	113	103
111	105	106	108
107	109	110	104
112	102	101	115

ALL ROWS, COLUMNS, DIAGONALS ADD TO CONSTANT 430

The magic square is a true magic square. All the rows, columns, and diagonals add to 430 in the sample square. Also, the four center squares, or the four corner cells (four squares making up a "corner") will also add to 430. In fact, the sum of any symmetrical group of four squares will equal the constant. In the sample square, the two middle cells in the top row (114 + 113) can be added to the two middle cells in the bottom row (102 + 101) to give constant 430 (114 + 113 + 102 + 101 = 430).

This program can be used to compute a magic square starting with ANY number, but I have decided to limit the range of starting numbers to 1 - 984 because of practical considerations. If you choose the number 984 as the starting number, the computer will develop a square using numbers in the range 984 - 999. Any starting number greater than 984 causes a rollover effect into four-digit numbers or more. I have decided to use numbers with three digits or less because of the limits on paper width. I would have to re-write most of the program to handle numbers with more than three digits. I feel the number range 1 - 984 is sufficient for practical requirements.

The program requests a number in the range 1 - 984. You type in the number and press the ENTER key. The program then causes the printer to draw a diagram with four rows and four columns. Then the computer does the math work and enters the numbers in the cells or squares. In the sample square, I used 100 as the starting number, and the computer developed a square using numbers in the range 100 - 115. The computers also figured the constant, in this case 430.

This makes an excellent demonstration program. You can load the program from tape, press the DEF SPACE keys to start the run and allow anybody to pick a number. You type in the number, and the computer responds almost immediately by drawing the diagram. This is fun to watch. After the magic square is finished, it can be torn off and handed out as a "personal magic square." It is a real intellectual challenge to determine how many ways numbers can be added to form the magic constant. The constant will be different for each starting number. I could set up a program to make a magic square that would add to a given constant, but this program would be of interest only to "magic square fiends." I have tried to develop a program that is of general interest, and if the reader wants to dig into magic squares, there are plenty of good books on the subject.

```
2Ø " "CLEAR
     : PAUSE "4 * 4 MAGIC SQUARE"
     : PAUSE "STARTING WITH ANY NUMBER"
3Ø INPUT "ENTER NUMBER (1-984)=";Y
     : IF Y>984PAUSE "INVALID ENTRY!"
     : GOTO 3Ø
4Ø TEXT
     : LF 1Ø
     : CSIZE 2
     : COLOR Ø
5Ø GRAPH
     : COLOR Ø
     : CSIZE 2
     : GLCURSOR (Ø,Ø)
     : SORGN
6Ø LINE (Ø,Ø)-(Ø,2ØØ)-(2ØØ,2ØØ)-(2ØØ,Ø)-(Ø,Ø)
7Ø LINE (Ø,Ø)-(Ø,15Ø)-(2ØØ,15Ø)-(2ØØ,1ØØ)-(Ø,1ØØ)
8Ø LINE (Ø,1ØØ)-(Ø,5Ø)-(2ØØ,5Ø)-(2ØØ,Ø)-(Ø,Ø)
9Ø LINE (15Ø,Ø)-(15Ø,2ØØ)-(1ØØ,2ØØ)-(1ØØ,Ø)
1ØØ LINE (1ØØ,Ø)-(5Ø,Ø)-(5Ø,2ØØ)-(Ø,2ØØ)
```

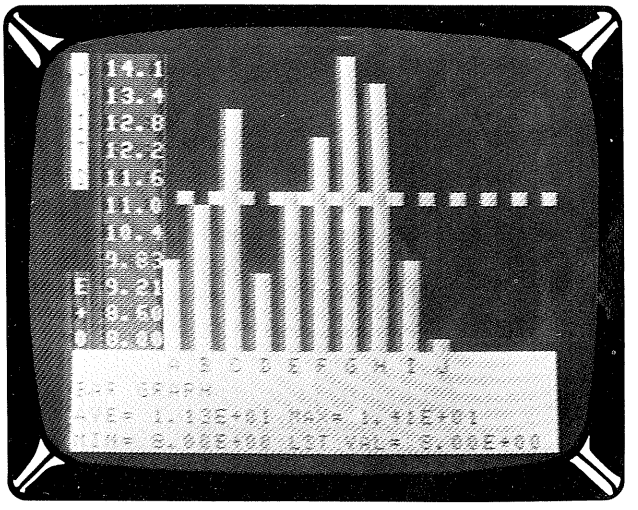
```
11Ø TEXT
     : COLOR 3
     : CSIZE 1
     : LF -1
     : USING "####"
12Ø LPRINT "4 * 4 MAGIC SQUARE STARTING WITH";Y
13Ø A=Y
     : B=Y+14
     : C=Y+13
     : D=Y+3
     : E=Y+11
     : F=Y+5
     : G=Y+6
     : H=Y+8
14Ø I=Y+7
     : J=Y+9
     : K=Y+1Ø
     : L=Y+4
     : M=Y+12
     : N=Y+2
     : O=Y+1
     : P=Y+15
     : Q=A+B+C+D
15Ø LF 4
     : CSIZE 2
     : LPRINT A;B;C;D
16Ø LF 1
     : CSIZE 2
     : LPRINT E;F;G;H
17Ø LF 2
     : CSIZE 2
     : LPRINT I;J;K;L
18Ø LF 1
     : CSIZE 2
     : LPRINT M;N;O;P
19Ø LF 2
     : CSIZE 1
     : LPRINT "ALL ROWS, COLUMNS, DIAGONALS"
2ØØ LPRINT "ADD TO CONSTANT";USING ",######";Q
21Ø CSIZE 2
     : COLOR Ø
     : LF 6
     : END
```

Bargraph

Dennis L. Hargens
3004 Linda Drive
Ennis, TX 75119

This program for an Extended Color BASIC Computer plots a bar graph of data that is entered from the keyboard. The vertical axis is labeled with the units of the graph and their exponent. The value of each line increment is also displayed, although this value must be multiplied by the exponent to obtain the correct value in relation to the graph. Each plotted value on the graph is labeled with a one character symbol on the horizontal axis. The title of the graph is displayed at the bottom of the graph. The title string can also be used to label the horizontal axis. The minimum, maximum, average,and last value are displayed at the bottom of the screen in exponential format.

```
1Ø DIM DA(13),LB$(13),UN$(8)
2Ø CLS
3Ø PRINT @ 226, "WHAT IS THE GRAPH'S NAME"
4Ø INPUT NAME$
5Ø Y=1
6Ø CLS
7Ø PRINT @ 135, "WHAT ARE THE UNITS"
8Ø PRINT @ 2Ø2, "OF THE GRAPH?"
9Ø PRINT @ 263, "PRESS <ENTER> AFTER"
1ØØ PRINT @ 33Ø, "EACH LETTER."
11Ø PRINT @ 389, "PRESS XX WHEN FINISHED"
12Ø INPUT UN$(Y)
13Ø IF UN$(Y)="XX" THEN 16Ø
14Ø Y=Y+1
15Ø GOTO 12Ø
16Ø Y=Y-1
17Ø CLS
18Ø PRINT @ 225, "HOW MANY DATA POINTS(1-13)";
19Ø INPUT DP
2ØØ REM DATA ENTRY
21Ø CLS
22Ø FOR IN=1TODP
23Ø INPUT "DATA POINT =";DA(IN)
24Ø INPUT "1-CHARACTER LABEL=";LB$(IN)
25Ø SUM=SUM+DA(IN)
26Ø IF IN=1 THEN 1Ø3Ø
27Ø IF MAX>DA(IN) THEN 29Ø
28Ø MAX=DA(IN)
29Ø IF MIN<DA(IN) THEN 31Ø
3ØØ MIN=DA(IN)
31Ø NEXT IN
32Ø AVE=SUM/(IN-1)
33Ø FAC=21/(MAX-MIN)
34Ø REM VERTICAL AXIS EXPONENT
35Ø CLS(Ø)
36Ø VA=MIN
37Ø IF VA<1Ø THEN 39Ø
38Ø VA=VA/1Ø
     :DE=DE+1
     :GOTO37Ø
39Ø IF VA>=1 THEN 41Ø
4ØØ VA=VA*1Ø
     :DE=DE-1
     :GOTO 39Ø
41Ø POKE 128Ø,5
42Ø IF DE>=Ø THEN 45Ø
43Ø POKE 1312,45
44Ø GOTO 46Ø
45Ø POKE 1312,43
46Ø EX=ABS(DE)
47Ø CO=EX+48
     :POKE 1344,CO
48Ø REM LINE INCREMENT VALUES
49Ø IC=(MAX-MIN)/1Ø
5ØØ FOR VO=1 TO 11
51Ø CZ=INT(VA)+48
     :SP=1347-(VO-1)*32
52Ø CO=(VA-INT(VA))*1Ø
     :CT=CO
     :CO=INT(CO)+48
     :CT=(CT-INT(CT))*1Ø
     :CT=INT(CT)+48
```



```
530 IF VO=1 OR ED=DE THEN 560
540 IF ED-DE>=2 THEN 570
550 POKE SP+1,46
    :POKE SP+2,CT
    :POKE SP-1,CZ
    :POKE SP,CO
    :GOTO 580
560 POKE SP-1,CZ
    :POKE SP,46
    :POKE SP+1,CO
    :POKE SP+2,CT
    :GOTO580
570 POKE SP+2,46
    :POKE SP+1,CT
    :POKE SP,CO
    :POKE SP-1,CZ
580 VA=MIN+VO*IC
    :ED=0
590 IF VA<10 THEN 610
600 VA=VA/10
    :ED=ED+1
    :GOTO 590
610 IF VA>=1 THEN 630
620 VA=VA*10
    :ED=ED-1
    :GOTO 610
630 NEXT VO
640 XF=12
    :XL=13
650 FOR LI=1376 TO 1407
660 POKE LI,143
670 NEXT LI
680 REM DATA PLOT ROUTINE
690 FOR PL=1TODP
700 NU=ASC(LB$(PL))
710 PO=1380+2*PL
720 POKE PO,NU
730 IF DA(PL)=MIN THEN 800
740 GY=INT((DA(PL)-MIN)*FAC)
750 FOR LA=21 TO (21-GY) STEP -1
760 SET(XF,LA,3)
770 SET(XL,LA,3)
780 NEXT LA
790 GOTO820
800 SET(XF,21,3)
810 SET(XL,21,3)
820 XF=XF+4
    :XL=XL+4
830 NEXT PL
840 REM VERTICAL AXIS LABEL
850 FOR DM=1 TO Y
860 NU=ASC(UN$(DM))
870 PO=992+DM*32
880 POKE PO,NU
890 NEXT DM
900 REM AVERAGE PLOT
910 GAV=21-INT((AVE-MIN)*FAC)
920 FOR AL=14 TO 62 STEP4
930 SET(AL,GAV,4)
940 SET(AL+1,GAV,4)
950 NEXT AL
960 REM DATA LIST
970 PRINT @ 384, NAME$
980 P$="AVE=##.##^^^^ MAX=##.##^^^^"
990 O$="MIN=##.##^^^^ LST VAL=##.##^^^^"
1000 PRINT @ 416, USING P$;AVE,MAX
1010 PRINT @ 448, USING O$;MIN,DA(DP)
1020 GOTO 1020
1030 MIN=DA(1)
1040 GOTO 270
```

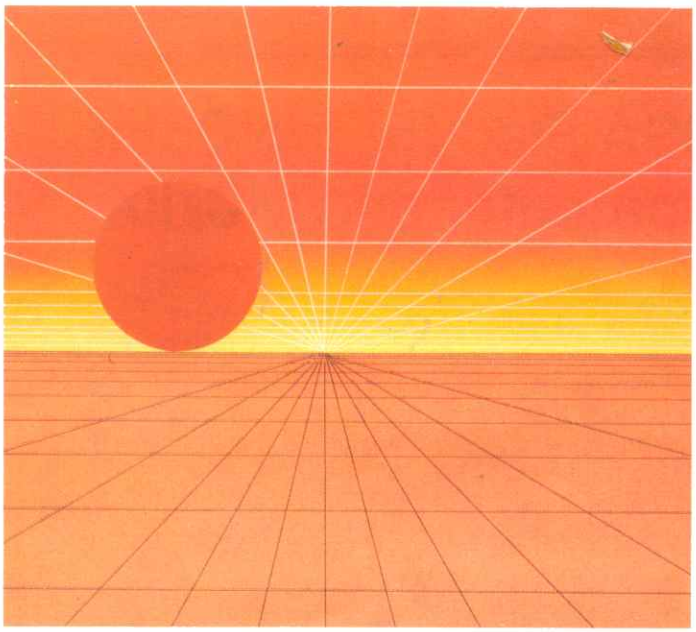
USA Flag for the Color Computer and MC-10

Calvin C. Epple
P.O. Box 86
Smithsburg, MD 21783

I'm having a ball with my TRS-80 Extended BASIC Color Computer and greatly appreciate the interesting articles, including programs, carried in your magazine. I have developed some interesting programs, too, and would like to share one with other readers of the *TRS-80 Microcomputer News*.

USA Flag is shown on this month's cover.

```
1 'USA BASIC BY CAL EPPLE 3,13,83
2 'FOR NON-EXTENDED AS WELL AS EXTENDED BASIC COCO'S
10 CLS
20 FOR X = 2 TO 61
     : FOR Y =2 TO 27
30 SET (X,Y,4)
     : NEXT Y,X
40 FOR X = 2 TO 61
     : FOR Y = 4 TO 24 STEP 4
50 SET (X,Y,5)
     : NEXT Y,X
60 FOR X = 2 TO 27
     : FOR Y = 2 TO 15
70 RESET(X,Y)
     : NEXT Y,X
80 FOR X = 4 TO 24 STEP 4
     : FOR Y = 4 TO 12 STEP 2
90 SET (X,Y,5)
     : NEXT Y,X
100 FOR X = 6 TO 22 STEP 4
     : FOR Y = 5 TO 11 STEP 2
110 SET (X,Y,5)
     : NEXT Y,X
120 PRINT@485, "AMERICA, THE BEAUTIFUL";
130 FOR X = 1 TO 59
140 READ N,L
150 SOUND N,L * .66
160 NEXT X
170 RESTORE
180 GOTO 180
190 DATA 147, 12, 147, 18, 125, 6, 125, 12, 147, 12,
     147, 18, 108, 6, 108, 12, 125, 12
200 DATA 133, 12, 147, 12, 159, 12, 170, 12, 147, 24,
     147, 12, 147, 12
210 DATA 147, 18, 125, 6, 125, 12, 147, 12,  147, 18,
     108, 6, 108, 12, 185, 12
220 DATA 180, 12, 185, 12, 193, 12, 159, 12, 185, 24,
     185, 12, 147, 12
230 DATA 193, 18, 193, 6, 185, 12, 176, 12, 176, 18,
     170, 6, 170, 12, 176, 12
240 DATA 185, 12, 170, 12, 159, 12, 147, 12, 176, 24,
     176, 12, 176, 12
250 DATA 176, 18, 159, 6, 159, 12, 176, 12, 176, 18,
     147, 6, 147, 12, 147, 12
260 DATA 159, 12, 176, 12, 147, 12, 185, 12, 176, 24,
     176, 12
```



Radio Shack®
TRS-80 Microcomputer News
P.O. Box 2910
Fort Worth, Texas 76113-2910

FIRST CLASS
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

TRS-80® Computer Camp

This summer let Radio Shack's qualified instructors introduce your child to the world of computers at a TRS-80 BASIC or a TRS-80 LOGO Computer Camp. One TRS-80 Color Computer will be available for every two campers to give each participant the opportunity for hands on experience. No prior computer knowledge is required.

TRS-80 LOGO Computer Camp

For Ages 8-11

The Color LOGO Language

Color LOGO provides unique features that are not found in any other LOGO language. There's the "doodle" mode and the fascinating "hatch" command that creates multiple turtles. The result is a LOGO for the TRS-80 Color Computer that will excite young minds and help them welcome the challenge of learning.

Computer Concepts for Young People

Radio Shack's Color LOGO is a revolutionary computer programming language for children. It is designed to help young minds grasp graphic relationships and develop problem-solving skills. Through manipulation of "turtle" graphics, youngsters learn to write simple programs. By doing so, the computer becomes a friendly learning tool, and youngsters gain insight into advanced concepts.

Sessions are Self-Paced . . . And Fun

Each child will progress at his or her own pace under the guidance of an experienced, professional educator. The curriculum, designed by educators, employs a play-oriented approach to encourage children to learn and use the computer.

TRS-80 BASIC Computer Camp

For Ages 12-15

Computer Concepts for Young People

Extended Color BASIC is an exciting and powerful programming language that youngsters can readily appreciate. Dynamic graphics, music and sound effects, and progressive programming techniques let students write simple problem-solving programs or create animated games.

Sessions are Structured And Not Intimidating

The five-day camp will consist of professionally-designed activities to keep the pace interesting and challenging.

Extended Color BASIC Language

Each camper will learn to use such BASIC features as assignment statements, looping, branching, subroutines, READ/DATA statements, editing, string packing, and more to create their own programs.

Introduction to LOGO

As an added bonus, each attendee will be introduced to Radio Shack's Color LOGO programming language. This will give the campers insight into a second computer language.

Radio Shack COMPUTER CENTER

Enroll now at your nearby Radio Shack Computer Center.