NORTHWEST OHIO 99'ER NEWS

Vol.4 No. 1 JANUARY 1986

This newsletter is published jointly by OH-MI-TI and New Horizons TI-99/4A Home Computer Users' Groups. Material may be reproduced without permission provided the author and source are acknowledged. For more information concerning TI Users' Groups in the Northwest Ohio area, contact:

Roger Biddle	Bill Sager
President, OH-MI-TI	President, New Horizons
218 Dillrose Dr.	612 Meadow Spring
Northwood s, Oh. 43619	Maumee, OH 43537
(419) 666-4945	(419) 893-7962


```
*********    TICOMM BBS  385-7484    *************
       *    SYSOP>>>BUD MILLS<<<    *
       *   !!!!!!! 24-HRS. !!!!!!   *
       *  SYSTEM OPERATING ON       *
       *  CORCOMPS NEW !!!!!!!!!!   *
       *  9900 MICRO EXPANSION SYS  *
       *                            *
       ******************************
```

THE NEWSLETTER STAFF

Kent Sheets	John Clulow
Roger Biddle	Paul Snyder
Earl Hoffsis	Bill Tiep
Marilyn Schafstall	Don Wollenbecker
Dave Burkett	Phil Bennis

WILLIAM SAGER
612 MEADOW SPRING RD.
MAUMEE, OH 43537

TOLEDO OHIO 1986

Dallas TI Home Computer Group
1221 Mosswood Place
Irving, TX 75061

PRESIDENTS CORNER

by Roger Biddle
OH-MI-TI

I would like to welcome everyone to the New Year. With the new year comes new club officers. They are as follows:
Exectutive Board

President	Roger Biddle
Vice President	Bob Peters
Secretary	Pat Hunsinger
Treasure	Dave Burkett

Board

Hank Alvaro
Dave Weldy
Peggy Williams
Jim Elfering
Ed M

I look forward to a great year for the club with these fine people, let them know your likes and dislikes, so they can be brought up at the board meetings and considered for the good of the club.

There will have been a board meeting prior to the January meeting to establish organization of the board members to their posts ex. club librarian, progam comm.,etc. Hopefully we will have a complete list of officers, and club members for you at the January meeting consisting of addresses, and phone numbers.

I was online on Compuserve the other night and there is all kinds of talk about the GRAM-KRACKER from Miller Graphics. This sounds like a real unique device creating new programs and applications for our computer. If anyone has one please bring it to the meeting to show us, I understand the first production run of the GRAM-KRACKER is already sold out.

I will try as your president to keep you informed as much as possible on new and exciting developements in software and hardware for our computer.

I hope everyone had a wonderful holiday season as I and my family did. Our next meeting is January 10, 1984. Please try to attend and lets get the new year for our club off to a fine start.

CONGRATULATIONS OH-MI-TI AND NEW HORIZONS MEMBERS ON YOUR THIRD ANNIVERSARY!

PRESENTATIONS

by Pat Hunsinger
OH-MI-TI

The January 10 meeting of OH-MI-TI will have two presentations 1. Don Turners will give an introduction to programming in the BASIC language. 2. Bob Peters will demonstrate the MBX system from Milton Bradley a (voice recognition system). Don gave the BASIC intro at NEW HORIZON'S in December with great success, so it should be a very good presentation. Bob received the MBX system for XMAS and has been playing baseball through voice recognition which I understand is a good program for voice recognition. Please try to attend and see these presentations they sound very interesting.

FOR SALE

TI99/4A with Books

SPEECH SYNTHESIZER

TI REMOTE CONTROLLERS

HOUSEHOLD BUDGET MGR PHM3007

TI LOGO I PHM3040

ADVENTURE PHM3041

EARLY LEARNING FUN

BOOK of HINTS for ADVENTURES

** MAKE AN OFFER **

Contact: Kent Sheets
836-7877

GRAPHX COMPANION AND COMPANION II

by Dave Burkett
OH-MI-TI

IF YOU'RE INTO USING YOUR TI-99/4A FOR CREATING GRAPHICS, THERE ARE A COUPLE OF REALLY GOOD GRAPHICS PACKAGES AVAILABLE. ONE SUCH PACKAGE (WHICH WAS DEMONSTRATED AT ONE OF OUR USER'S GROUP MEETINGS), AND AURGUABLY THE BEST AROUND COMES TO US FROM THE TALENTED PEOPLE IN AUSTRALIA. I'M SPEAKING OF GRAPHX, OF COURSE. SOME OF THE MOST SOPHISTICATED GRAPHICAL REPRESENTATIONS THAT I'VE SEEN TO DATE HAVE BEEN DONE USING THIS SOFTWARE, AND INDEED MANY OF THE LITTLE CARTOONS, ETC. USUALLY FOUND SPRINKLED AROUND THE PAGES OF THIS NEWSLETTER HAVE BEEN CREATED BY GRAPHX.

BUT GRAPHX ISN'T WHAT I WANT TO TALK ABOUT. IN MY RECENT TRIP TO THE CHICAGO TI FAIRE, I PURCHASED TWO OF THE BEST PIECES OF SOFTWARE IN MY LIBRARY, COMPANION AND COMPANION II. AS YOU'RE AWARE, IF YOU'VE USED ANY GRAPHICS PACKAGE IT CAN BE A REAL DRAG CREATING AND RE-CREATING THE SAME CHARICATURES OVER AND OVER IF YOU NEED TO. THAT'S WHERE COMPANION AND COMPANION II COME IN HANDY. THESE TWO SOFTWARE PACKAGES FROM ASGARD OF ROCKVILLE, MARYLAND ARE REALLY INDISPENSABLE TO THE SERIOUS GRAPHICS FREAK. BETWEEN THEM THEY HOLD 24 DIFFERENT TYPE FONTS FOR CREATING HEADERS, TITLES, ETC., AND 14 "CLIPART" COLLECTIONS. "CLIPART" IS THE TERM USED FOR ALL OF THE USEFUL PICTURES OF OBJECTS, LOGOS, AND OTHER ITEMS THAT ARE USEFUL FOR SPICING UP YOUR ARTWORK. I HAVEN'T GONE THROUGH ALL OF THE DIFFERENT PRINT FONTS YET, BUT WHAT I HAVE USED IS REALLY IMPRESSIVE.

IN ADDITION TO THE FONTS AND CLIPART PROVIDED, THE AUTHORS HAVE ALSO INCLUDED SEVERAL SCREENS WHICH DEMONSTRATE THE USE OF COLOR, CERTAIN DRAWING TECHNIQUES, AND TO CONTAIN ART THAT IS SOMEWHAT TOO LARGE TO EASILY STORE IN A CLIPBOARD.

CONT. PG 4

NEW HORIZON NEWS

by Bill Sager

As my fingers stumble over the keys of TI-Writer to begin another New Horizons News column I can not help but think back over the past three years that our group has existed. The people, the progress, the changes, the joys, the frustration. Mostly the people, our members, come to mind. Because, without the dedication of a number of individuals there would be no New Horizons Computer Users Group.

On our third anniversary it seems fitting to pay tribute to two of those members. They were the organizers of the group and have remained active, involved members. The contributions made by these two people would stretch pages and then probably not include everything that they have done for our users group. I would venture to say that there are very few members who have not been helped or benefited directly from either one of these individuals. By now, most of you know that I refer to John and Norma Clulow. So, to John and Norma --- THANK YOU VERY MUCH on behalf of ALL the New Horizon members.

If you have not yet paid for your 1986 membership; please do so at the January meeting or you can mail a check to Treasurer Earl Hoffsis. All of those who paid by December 20 will have their names included in the drawing for the $50 software package, $15 cash, $8 worth of club software and $5 worth of club software. This will be the last newsletter for those who have not paid. More than ever, your users group membership is the key to unlock your computers potential. We are planning some things for this year to help you get more from your TI computer. We also will keep you current on the latest developments in software, hardware and freeware.

Speaking of freeware, there is a move underway to change the name of this means of marketing software to shareware or fairware. Apparently, it is thought that too many users have concentrated on the free part of the word and have not made even a small donation to the authors. Personally, I am not sure that changing the name will make much difference as this becomes a matter of ethics for most people. The following definition of freeware appeared in the newsletter of the K-Town 99'ers, Knoxville, TN: "Free to View, Free to Copy, Free to Distribute. Free to PAY the developer what you think the program is worth. AND an opportunity for you to prove the strength of your character."

Along this same line, our group has made donations to some program authors for items we have been able to distribute for them. There have been different arrangements worked out with each author as sometimes the club has received a small part of the proceeds and other times we have done this simply to benefit our membership by getting above average quality, low cost software to you. Since this has proven to be very popular with our members we will continue to do this whenever possible. It was by special arrangement with Barry Traver that we were able bring his first edition of the GENIAL TRAVelER Diskazine (magazine on disk) to you. We are the only users group that was permitted to do this. Except for the bonus disk, the diskazine is not to be copied. The price was so reasonable that we trust you will respect this.

A good source of quality DS/DD diskettes has enabled the group to purchase a large quantity and they will be available at the meeting for 75 cents each for members only. Another reason to join and attend!

A big thanks to members, Mike Amundsen and Don Turner for their informative and interesting presentations in December. Don has made the program he wrote and demonstrated available to the group and you will be able to obtain it at the January meeting. Good news for those interested. Of course, Mike's routines were available on disk in December.

There was an impressive assortment of food and snacks for our holiday enjoyment last month. I overheard many compliments that those who brought the goodies may have missed. I hope you enjoyed the gourmet delights! Thank you food donors - Earl Hoffsis, Diana Echler, James Brett, Peter Dobrusky, Linda Rawlins, Marilyn Schafstall, Phil Bennis, Norma Clulow, Connie Sager, Ron Gries, Bill Tiep, Margaret Dixon, and Wayne Zachrich.

The new officers for this year should be inspired by the vote of confidence they received. One hundred percent yeas and no nays. OK guys, the 1988 White House field is wide open. What ya say? We'll call it the "Computer Party". Anyway, the officers are: President - Bill Sager, Vice President - John Clulow, Treasurer - Earl Hoffsis, and Secretary - Russ Lee.

Our sincere appreciation to retiring officers Bob Dilworth and Paul Sneider. Also, thanks to Lida Lee who has hosted the check-in and name tag table the past year and has agreed to continue doing so. Plans are to have name tags for everyone as soon as possible and also a new up-to-date membership roster to pass out. This may be in February to make it complete for the late joiners.

For the January meeting we anticipate having two or three new disks of items you may purchase at the usual low prices. The presentations will be by Don Turner with his second segment on BASIC; John Clulow showing us "Special Tricks in TI-Writer"; and Mike Amundsen doing a mini demo of his latest creation "Merge File Editor".

There are several more TI users group sponsored major events coming up in the next several months. One is in March in Los Angles and then three on the east coast this spring. I don't have all the details yet but if anyone is interested, let me know and I'll try to get more information or a contact for you. Not bad for a computer dropped well over two years ago! Although not a TI event, but still of interest locally is the Hamfest and Computer Show that will be held March 16 at the Lucas County Recreation Center. Last year it was about 50/50 computer and amateur radio. Both exhibit halls were filled. This is a place for electronic bargains, both new and used.

The New Horizon meeting is January 11th at 2 PM, Unity Church, 3535 Executive Pkwy, Toledo, just north of the Westgate Shopping Complex. Kick off the new year right and plan on being there!

CONT. from PG 2

TO SUM UP, IF YOU HAVE GRAPHX AND DON'T HAVE AT LEAST ONE OF THESE COMPANION PROGRAMS, THEN YOU JUST DON'T HAVE THE FULL POWER OF GRAPHX AT YOUR COMMAND. AT LESS THAN $10.00 EACH, THEY ARE A STEAL.

A FORTH TUTORIAL PART ONE

One thing many of us have been looking for is an easy-to-read introduction to the programming language FORTH -- after all it is FREE to all Users Group members!

Les Koth, a recent New Horizons member, has offered to write some introductory articles on FORTH during the comming year. The first of these appears below.

Even if you know nothing whatever about FORTH -- if you have just heard the word at meetings -- you should have no trouble understanding Les' great introduction.

For those who may not have had the pleasure of meeting Les (he is in the National Guard and can't always make the Saturday meeting), he is Chief Quality Contol Engineer with LOF. Les has a B.S. in physics and math and an MBA. Prior to working for LOF, he was manager of shop operations at Gt's Newark OH plant for 11 years (where they make silica from which silicon computer chips eventually are made).

We are very happy Les has taken the time to make such an excellent contribution to our newsletter, and hope to have him give us a FORTH demonstration "live" at a future meeting.

GOING FORTH
by Les Koth

New Horizons

Many of us have heard of FORTH since TI released its public domain version of this exciting language but if you are like many people you may not have taken the time to really learn and understand it. There are lots of good reasons not to take the time to learn FORTH, including the fact that the TI version is largely unsupported (if you know of anyone who does please let me know!). There are however many more reasons why you should learn FORTH. This article will present an overview and look at the pro's and con's of FORTH as a problem solving tool. In subsequent articles we will look at Wycove FORTH and TI FORTH--both of which are available to run on your TI-99. For those of you who feel limited by hardware (i.e. no disk drive or PE box) keep in mind that about $20 worth of parts can allow you to build in the 32k Matchbox memory expansion (Vol. 3 No. 5 May 1985, this newsletter)into your console and with a SUPER CART (published in MICROpendium June 1985 and demonstrated by Ron Gries at the Nov 85 meeting) with Mini Memory GROM (about $15 in parts) and a tape recorder you too can use FORTH!

First of all some comments are in order about programming languages in general. Regardless of whether you select a convenient high-level language like BASIC, FORTRAN, PASCAL or a "low-level" language like Assembly Language the basic hardware capability of your computer remains the same. The machine processes machine language statements one at a time at the clock rate of the processor. Even the Artificial Intelligence languages such as LISP and its subset LOGO must eventually execute as machine language statements on the microprocessor used in your computer. So, when we talk about how efficient or "fast" a programming language is we are really talking about how convenient it is to state a problem for solution. This is what any programming language is--a way to state problems for solution. It would be nice if the environment in which we make our problem statements would let us quickly test small portions of the problem statement and at the same time allow us to "design" the problem statement in the most general terms. Finally, it would be nice if the completed problem "design" would execute as fast as possible on our computer. These are some of the reasons to consider FORTH.

Until I found out that FORTH was available for the TI-99 (in at least two versions) I assumed that it was just another fancy high-level language someone had put together because they didn't like BASIC, or FORTRAN. That was before I decided to find out about it. Upon investigating FORTH, I found that it had very little in common with BASIC or FORTRAN and that it seemed to be the answer to my prayers in terms of getting the most out of my TI-99. When I first got my system I purchased Multiplan, Editor/Assembler, Extended Basic and Minimemory. I was determined to get the most out of our computer and I was going to try every package I could get my hands on for a reasonable price. Well, Mulitplan is an outstanding spreadsheet but did you ever fill up a sheet and then make a recalculation? It takes a full 45 minutes for the computer to process one recalc on a full sheet!!! Not very fast if you're in a hurry to do "what ifs". I've written a lot of problem statements in BASIC and had to put up with long hours of debugging even though I try to be very structured and modular in my designs. Extended Basic although more powerful in terms of the statements one can make seems to run at least as slow as Multiplan. I was afraid that I was going to have to learn Assembly Language in great detail in order to fully use my TI at the lightning fast speed that I know a 16-bit processor is capable of.

So what is FORTH? It is more than just another set of instructions to program your computer. It is a high-level computer language but is also an assembly language, an operating system and very specifically a philosophy on how to structure and design solutions to problems. I will borrow some comments from Michael Amundsens article in the April 85 newsletter on PULSAR. Michael came to some of the same conclusions that I have come to regarding speed and efficiency of assembly programming. It takes forever to learn assembly language and forever to write it but it does run very fast once you have it debugged. Michael appropriately came to the conclusion that what is needed is a set of "portable" utility routines that can be used by anyone in designing a problem solution. Interestingly enough this is exactly one way of viewing what FORTH does for you. It gives you access to a rich variety of "primitives" or "words" which are actually calls to a modular set of assembly language routines.

CONT. PG 5

CONT. from PG 4

GOING FORTH

In FORTH you define your own applications in terms of any of the basic words in the FORTH dictionary just as you might do in TI LOGO. You can then build on previous words that you have defined until--at the highest level--you have one word which is your application. One advantage of defining your own words in the FORTH operating system is that you can try them out immediately after defining them. You don't have to go from source code to object code and you don't have to wait until most of the program is written as you might in BASIC. The FORTH operating environment is similar to the Immediate mode in TI BASIC. You type in commands and they are executed immediately. This is another key advantage of FORTH--you can test small portions of your application thoroughly and immediately.

FORTH is a compiled language as opposed to an interpreted language which means that in some cases it can run up to 40 times (that was not a typo--forty) faster than an equivalent Extended BASIC program. Don't always expect that type of performance though. Typically if an aplication has a moderate amount of arithmetic it may be only two to five times faster than an equivalent BASIC program. One of the reasons for this increased speed is that basic FORTH does not have any floating point math routines. This means that everything must be done in integer math. However, with our TI-99 we have access to console ROM math routines which are floating point and both TI and Wycove FORTH give ready access to these.

FORTH is what is known as a "threaded interpretive language" which means that the "words" in the dictionary are "threaded" together by pointers (much like Michael Amundsens PULSAR) which tell you where to find the next word to be executed. The advantage to this type of structure is that it can be very compact in terms of memory requiremets for a given application. So much so in fact that a given application could require less memory than an equivalent assembly language program! This is possible because a given application consists of pointers which call the same basic set of "primitve" words or assembly language routines rather than executable code. The cost of this type of structure is paid in terms of speed. Most FORTH applications run somewhat slower than comparable assembly language programs because time is spent moving through data pointers.

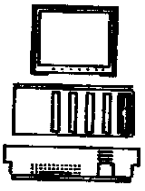
Two more features of FORTH bear discussion. FORTH is a stack based language. For those of you who own Hewlett-Packard calculators you probably are quite comfortable with the notion of a "stack". The stack is a "structure" in memory which handles numbers. To do arithmetic operations you place numbers on the stack and then apply operators to get the result. In practical terms for those of you who aren't comfortable with the notion of a "stack" (isn't that where you have to use that crazy Reverse Polish Notation?) what it boils down to is this--you don't have to worry about parentheses! I have spent countless hours debugging BASIC programs where I had nested parentheses and left one crucial one out! With the stack structure you don't ever have to worry about them again. One of the nice things about FORTH as a high-level language is that most of the FORTH primitives that work on the stack automatically prepare it for most of the words which would be used serially in an application. This means that in many cases you don't have to worry about the stack, just define your application and get your results.

There is one final aspect of the FORTH operating system that should be discussed. FORTH uses the concept of "virtual memory". This means that with the exception of occasional automatic disk accesses the user is unaware of memory limitations. FORTH uses a term called "blocks" or "screens" to store applications in. These blocks are read in as needed to retrieve data or add words (applications) to the dictionary. Buffers in the system automatically refresh and update screens that are being worked with. With the advent of the RAM DISK for our TI-99, this aspect of FORTH could be especially useful! Imagine having 192K of virtual memory at your fingertips through the FORTH operating system!

Lest I make FORTH sound all rosy with no problems let me mention a few of the things that may be drawbacks. There are relatively few error detection routines in FORTH. This means that you are fully in the drivers seat! It is not infrequent to have your system crash while you are learning what FORTH can do for you. Of course you can add your own error handling words if you feel the need. Because FORTH has a different structural concept it is also more difficult to learn than BASIC for instance.

Let me quit this article with some comments on some of the specific features that FORTH has on the TI-99. One of the more interesting features is the implementation of a 64 column display. Full screen editing is a common feature, full graphics, sprite, sound and speech routines are available to make you feel almost right at home if you have been programming in BASIC. FORTH has provisions for writing 9900 assembly code directly in the FORTH environment for those of you who require assembly speed in certain applications. The Wycove version is fully reconfigurable--you can put it any where in memory that you want, change startup parameters, change size and location of buffers and other user variables. Words which give direct access to ROM, GROM and DSRs are standard. The TI version has two interesting features which allow a split video display--the bottom half in text mode and the top half in bit-map mode which allows you to give commands normally in the lower portion of the screen and do bit-map graphics on the top. As you can see there is a lot of power and potential in using the FORTH programming environment! Next time we'll talk about some specific applications and using FORTH.

TI 99/4(A)



THE SUPER SYSTEM

HAPPY BIRTHDAY NEW HORIZONS AND OH-MI-TI COMPUTER USERS GROUPS 3 YEARS OLD - JANUARY 1986

MERGE FILE EDITOR Makes Programming Easier

Michael C. Amundsen

New Horizons

TI EDITOR IS GOOD, BUT

In the time I have spent writing TI BASIC and XBASIC programs, I have come to appreciate the TI Line Editor built into the console. Of all the home computers, TI's Line Editor is about the best I've worked with. Few computers offer the easy editing of a single line (typing NUM XXX or EDIT XXX and using arrow keys, etc.) or the global resequencing of program lines (great when you have to insert a line later) that the TI Line Editor has. In fact, in many machines, you need to use a word-processor to generate your original textfile for the BASIC program (goodbye automatic line-numbers!).

There are some times when I could use some more flexibility than the current TI Editor offers, though. There are four editing actions that I often need, but are not allowed by the built-in console editor. They are: 1) delete a series of lines (say a whole subroutine); 2)copy a series of lines to another file for use in other programs; 3) move a series of lines to another area in the same program (for example, move all data statements to the end of the program); and 4) delete only the REM lines to save memory space once the program is completed.

To meet my needs for a more flexible editor (and my need to continue to write programs!), I wrote a progam called MFE (Merge File Editor) that allows the editing actions I described above. This program works only on XBASIC's MERGE Format files and requires a disk drive, expansion Memory and, of course, the XB cartridge. Below is a run-down of the capabilities of this small, but powerful programming aid.

WHAT THE MFE CAN DO

The MFE is great for doing little "spot-editing" on your programs. It alows you to copy or delete any line or sequence of lines in your program, delete only the comment lines, and resequence any line or group of lines including moving a group of lines from one part of the program to another. All these functions can be done on any BASIC or XBASIC program as long as it has been SAVEd in XBASIC's MERGE format.

DELETE-ing lines

If you suddenly realize that the subroutine you just wrote is a duplicate of some other lines in your program, you could use the built-in editor to erase each line, one at a time (and sit and wait around!) or you could use the MFE to do it all at once.

MFE asks you what the starting and ending lines to delete are and then creates a new program file with the offending lines removed.

COPY-ing Lines

I often discover that the subroutine I need has already been written in some other program. Instead of getting the printout and sitting at the console typing the thing in again, I just use the MFE to copy the desired lines from the original program into another file for use in my new project. This saves time, effort and reduces the chance of typing errors in transfering the routine.

Deleting REM Lines

I tend to write a lot of comments in my programs as I am designing them. It helps me remember where I am headed when I come back to the project later on. But these comments use up precious memory and need to be removed to improve the speed of the program. I use the MFE to delete all 'REM' and '!' comment lines from my completed programs.

RESEQUENCING Lines

This is by far the most handy of the MFE functions. It allows me to outline a specific set of lines (say 1050-2015) and to resequence them using any starting line number (say 3000).

This may not seem handy at first, but I have come to love this feature of MFE. Below are some examples of the use of resequencing to help imporve programs:

1 - KEEPING THINGS NEAT

I like to keep things easy to read and edit when I write a program. I try to start all major routines with similar line numbers like 1000, 2000, 3000, etc. and I try to keep all line numbers in increments of 10.

When I am de-bugging, however, things get a bit messed up. discovering the need to add an extra line can mess up the line numbers, and using the TI editor to resequence them can botch up my 1000, 2000, 3000 sections, too!

I can use MFE to fix this, though. I can tell MFE to resequence lines 1000-1135 in increments of 10 (or 5, 20 ,etc) starting at 1000. No other lines will be affected and every jump-reference (GOTO, GOSUB, etc.) will be adjusted if needed. Handy, eh?

2 - MOVING THINGS AROUND

The MFE can also move entire sections of code from on part of the program to another. How amny times have you discovered you have just written some program code underneath a XBASIC Subprogram? The program won't run because all Subprograms must be at the end of the program code! How about when you wish you had put that subroutine at the end of the file instead of the middle? Or how about wanting to put all your DATA statements in one section instead of scattered throghout your program? Do you delete the code and write it all again in the proper place? Not if you have MFE.

With MFE you can move any lines of code by just changing the starting address of the resequencing. For example, say I wanted to move the DATA statements now at lines 350-460 down to the end of the file at around 1500. All I need to do is tell MFE to resequence starting at 350 and ending at 460 and start the new line numbering at 1500 in increments of 10. MFE does the rest!

CONT. PG 7

CONT. from PG 6

MFE DISK AVAILABLE

MFE has become a standard tool in my programming arsenal and I highly recommend it for anyone who does alot of BASIC and XBASIC programming.

A program disk including on-line instructions will be available for $5 at the January New Horizons Club Meeting or by contacting me at:

Michael Amundsen c/o SubFile99
POB 533
Bowling Green, Ohio 43402

CIS: 71706,625
STC: TI5361

As usual, if you have discovered other uses/needs for a program such as this, I'd love to hear from you!

KEEP IN TOUCH - MA

The Best of Steve Patterson

by Steve Patterson
New Horizons

I would like to thank anyone who has bought my disk "The Best of Steve Patterson". I hope you enjoyed running and playing with the programs. Because of the large amount sold, all that I had, of the disk at the last New Horizons meeting I have decided to run off another ten copies for the next meeting.

I hope to get to the OH-MI-TI meeting this month. The price of the disk, in case you forgot, is only $2.00 so even if you do not like any of the programs (which is impossible) you can just sweep the disk and you still have a double-sided (flippy) blank disk.

ADVENTURES IN BASIC - #1

by Steve Patterson
New Horizons

As I am sure you have already heard or encountered, the worst problem in console BASIC is the absence of a DISPLAY AT command. There is a CALL HCHAR which prints a single variable on the screen, but what about an entire statement? And the CALL HCHAR statement requires a NUMBER -- the ASCII code of the character -- so you also have to worry about how to change a string into a number to use this statement.

Well, here is a program which takes a string variable (such as "Steve Patterson") and takes each character, changes it to a numeric variable (the ASCII code) and prints it on the screen wherever you want it to be!

```
1000 FOR T=1 to LEN(A$)
1010 B$=SEG$(A$,T,1)
1020 CODE=ASC(B$)
1030 CALL HCHAR(ROW,COL+T-1,CODE)
1040 NEXT T
1050 RETURN
```

Place this at the end (or start for better speed) of your program, and when you wish to print a statement or a line or whatever, just assign the string to A$, set ROW and COL as the row and column on the screen where you want to start. Be careful that the number of characters in the string will not run off the right side of the screen or you will get an error. Then after assigning the different variables, you GOSUB 1000.

```
100 ROW=12
110 COL=9
120 A$="HAPPY NEW YEAR"
130 CALL CLEAR
140 GOSUB 1000
150 END
```

Will "Display At" the message in the center of the screen.

If you want to print a number -- 1986 -- you must first make it a STRING! This can easily be done using the STR$ statement.

```
100 ROW=12
110 COL=6
120 C$=STR$(1986)
130 A$="HAPPY NEW YEAR "&C$
140 CALL CLEAR
150 GOSUB 1000
160 END
```

ADVENTURES IN BASIC - #2

by Steve Patterson
New Horizons

In this article, I will show you a simple but useful program. Useful, that is, if you have a disk system -- it is one of the functions of many good Disk Manager programs like DM-1000 (which is where I got the idea for my program).

A disk sweep program, in effect, gives you a "clean, initialized disk" by simply initializizing the "catalog" portion of the disk. Because it does not actually erase or re-format all the sectors, it can be a lot faster than re-formatting all the sectors.

Using a program like DM-1000, which is in machine language, you can "erase" an entire disk in about two seconds! While mine does not work that fast, it gets the job done. The thing that's different about my disk sweep program is that it is written in console BASIC!

The program itself works very much like a catalog. It starts by reading the first sectors of the disk: the ones containing the disk name and file names. Exactly which sectors those are I don't know, but they are around the first few.

Anyway, the program reads all the names of the files, and as it's reading them, it deletes them off the disk. For it to delete the entire disk it takes about four passes. Why it does not run and get it over with I do not know, but maybe someday I will write an updated version that only takes one time to delete an entire disk.

Possibly, now that I come to think about it, it may even take more than four passes for a double-sided double-density disk. (I do not know because I only have a single sided drive.)

PROGRAM ON Pg. 8

Cont. from PG 7

DISK SWEEPER

```
100 CALL CLEAR
110 PRINT "       DISK SWEEP
ER             ERASE AN ENTIR
E DISK          IN LESS THAN
A MIN."
120 PRINT
130 PRINT "     STEVE PATTER
SON"
140 FOR T=1 TO 5
150 PRINT
160 NEXT T
170 PRINT "PLACE DISK IN DIS
K DRIVE #1   PRESS ANY KEY T
O DELETE"
180 CALL KEY(0,K,S)
190 IF S=0 THEN 180
200 CALL CLEAR
210 R=1
220 OPEN #1:"DSK1.",INPUT ,I
NTERNAL,RELATIVE
230 INPUT #1:A$
240 B$=A$
250 GOSUB 430
260 R=R+2
270 INPUT #1:B$
280 IF B$="" THEN 320
290 GOSUB 430
300 DELETE "DSK1."&B$
310 GOTO 270
320 CLOSE #1
330 CALL CLEAR
340 PASS=PASS+1
350 IF PASS=5 THEN 390
360 PRINT ">>> PASS #"
370 CALL HCHAR(23,13,PASS+49
)
380 GOTO 210
390 PRINT "    COMMAND COMPL
ETED            PRESS ANY K
EY"
400 CALL KEY(0,K,S)
410 IF S=0 THEN 400
420 END
430 IF R=24 THEN 510
440 FOR U=1 TO LEN(B$)
450 C$=SEG$(B$,U,1)
460 A=ASC(C$)
470 CALL HCHAR(R,U+3,A)
480 NEXT U
490 R=R+1
500 RETURN
510 PRINT
520 R=23
530 GOTO 440
```

New Horizons Next Meeting

Saturday January 11 2:00

Special Computer Drawing!

In addition to the three presentations below, we'll have some new disks including Don Turner's great "Phonebook" program demonstrated at the last meeting. A version of TK Writer is available that allows the Show Directory (SD) command, and that file will be on the Don Turner disk for TK Writer users. A second disk will be an Ext. BASIC Loader for Editor/Assembler

Boxes of 10 blank generic disks will be available for $7.00

PRESENTATION #1: Don Turner will give "mini-lesson" #2 in BASIC programming.

PRESENTATION #2: Mike Amundsen will demonstrate his Merge File Editor program (See pg. 6). Copies will be available.

PRESENTATION #3: John Clulow will demonstrate some time saving techniques to use with TI Writer.

BUY A COMPUTER AND BE ORGANIZED THE AD SAID - I HAVE THE COMPUTER 3 YEARS NOW AND AM STILL NOT ORGANIZED

1986 T.M.R.A.

HAMFEST and COMPUTER SHOW

SUNDAY, MARCH 16

LUCAS COUNTY REC CENTER

2901 KEY ST. MAUMEE, OH

This is always a great event, so plan now to attend.

Anyone interested in setting up a TI Users Group Table?

CAN A PERSON UNDER 18 USE XBASIC AND XMODEM?

SOME UNUSUAL COMPUTER LANGUAGES...

Copy provided by Rod Cook, Newark OH - Member OH-MI-TI

(This article was keyed-in from a DEC Users Group newsletter. 75116,3407)

==========

Languages NOT included in the Commercial Language SIG or the Languages and Tools SIG

by
Doug Bohrer
Bohrer and Company
Near Chicago
and
Ted A. Bear
NCA Corporation
In the heart of Silicon Valley
and
A Usually Reliable Source
Digital Equipment Corporation
Somewhere in New England

APL, BASIC, COBOL, FORTRAN, PASCAL, RPG ... these programming languages are well known and (more or less) loved throughout the computer industry. There are numerous other languages, however, that are less well known yet still have ardent devotees. In fact, these little known languages generally have the most fanatic admirers. For those who wish to know more about these obscure languages -- and why they are obscure -- we present the following catalogue.

C-
This language is named for the grade received by its creator when he submitted it as a class project in a graduate programming class. C- is best discribed as a "low level" programming language. In general, the language requires more C- statements than machine code instructions to execute a given task. In this respect it is very similar to COBOL.

DOGO
Developed at MIOT (Massachusetts Institute of Obedience Training). DOGO hearlds a new era of computer literate pets. DOGO commands include SIT, HEEL, STAY, PLAY_DEAD and ROLL_OVER. An innovative feature of DOGO is "puppy graphics", a small cocker spaniel that occasionally leaves deposits as it travels across the screen.

FIFTH
FIFTH is a precise mathematical language in which the data types refer to quantities. The data types range from CC, OUNCE, SHOT and JIGGER to FIFTH (hence the name of the language), LITER, MAGNUM and BLOTTO. Commands refer to ingredients such as CHABLIS, CABERNET, GIN, VERMOUTH, VODKA, SCOTCH, BOURBON, CANADIAN, COORS, BUD, EVER-CLEAR and WHAT_EVERS_AROUND.

The many versions of the FIFTH language reflect the sophistication and financial status of its user. Commands in the ELITE dialect include VSOP, LAFITE and WAITERS_RECOMMENDATION. The GUTTER dialect commands include THUNDERBIRD, RIPPLE and HOUSE_RED. The GUTTER dialect is a particular favorite of frustrated FORTH programmers who end up using this language.

LAIDBACK
This language was developed at the Marin County Center for T'ai Chi,

Mellowness and computer programming (now defunct), as an alternative to the more intense atmosphere in nearby Silicon Valley.

The center was ideal for programmers who liked to soak in hot tubs while they worked. Unfortunately few programmers could survive there because the center outlawed Pizza and Coca-Cola in favor of Tofu and Perrier.

Many mourn the demise of LAIDBACK because of its reputation as a gentle and non-threatening language since all error messages are in lower case. For example, LAIDBACK responded to syntax errors with the message:

> "I hate to bother you, but I just can't relate to that.
> Can you find the time to try it again?"

LITHP
This otherwise unremarkable language is distinguished by the absence of an "S" in its character set. Programmers and users must substitute "TH". LITHP is said to be useful prothething litht. This language was developed in San Francisco.

REAGAN
This language was also developed in California, but is now widely used in Washington D.C. It is the current subset ot the international bureaucratic language known as DOUBLESPEAK. Commands include REVENUE_ENHANCEMENT, STOCKMAN, CAP_WEINBERGER, MALCOMB_BALDRIDGE, CABINET, CHOP_WOOD, LAXALT and SCENARIO. WATT and BURFORD have been removed from the commands while there is a current effort to add MEESE.

The operating systems used is NEW_RIGHT and the designated memory is THE_RANCH. The compile SCENARIO is a compile with NANCY followed by a link with BONZO resulting in a SNOOZE. COMMIES (program bugs) are removed with the GRANADA command.

A REAGAN program commences with LANDSLIDE and terminates with SENILITY.

RENE
Named after the famous French philosopher and mathmetician Rene DesCaters, RENE is a language used for artificial intelligence. The language is being developed at the Chicago Center of Machine Politics and Programming under a grant from the Jane Byrne Victory Fund. A spokesman described the language as "Just as great as dis (sic) great city of ours."

The center is veryy pleased with progress to date. They say they have almost succeeded in getting a VAX to think. However, sources inside the organization say that each time the machine fails to think it ceases to exist.

SATRE
Named after the late existential philosopher, SATRE is an extremely unstructured language. Statements in SATRE have no purpose; they just are. Thus SATRE programs are left to define their own functions. SATRE programmers tend to be boring and depressing and are no fun at parties.

SIMPLE
SIMPLE is an acronym for Sheer Idiot's Monopurpose Programming Linguistic Environment. This language, developed at Hanover College for Technological Misfits, was designed to make it impossible to write code with errors in it. The statements are, therefore, confined to BEGIN, END and STOP. No matter how you arrange the statements, you can't make a syntax error.

SLOBOL
SLOBOL is best known for the speed, or lack of it, of the compiler. Although many compilers allow you to take a coffee break while they compile, the SLOBOL compiler allows you to travel to Columbia to pick the coffee. Forty-three programmers are known to have died of boredom sitting at their terminals while waiting for a SLOBOL program to compile.

VALGOL
From its modest beginnings in Southern California's San Fernando Valley, VALGOL is enjoying a dramatic surge of popularity across the industry.

VALGOL commands include REALLY, LIKE, WELL and Y*KNOW. Variables are assigned with the =LIKE and =TOTALLY operators. Other operators include the California Booleans, AX and NOWAY. Repetitions of code are handled in FOR - SURE loops.

Here is a sample program:

```
     LIKE, Y*KNOW (I MEAN) START
IF PIZZA            =LIKE BITCHEN AND
GUY                 =LIKE TUBULAR AND
VALLEY GIRL         =LIKE GRODY**MAX(FERSURE)**2
THEN

     FOR I =LIKE 1 TO OH*MAYBE 100
          DO*WAH - (DITTY**2)
          BARF(I) =TOTALLY GROSS(OUT)
     SURE

     LIKE BAG THIS PROGRAM
     REALLY
     LIKE TOTALLY (Y*KNOW)
     IM*SURE
     GOTO THE MALL
```

VALGOL is characterized by its unfriendly error messages. For example, when the user makes a syntax error, the interpreter displays the message:

GAG ME WITH A SPOON!!

(This article first appeared in the APL SIG newsletter THE SPECIAL CHARACTER SET (D. Bohrer, Editor) and has gained steam ever since.)

From: Boise 99'er Dec. 1985 pg 4

****

```
Enlarged Print      -- CTRL U SHIFT N CTRL U
Cancel Enlarger     -- CTRL U SHIFT T CTRL U
Condensed Print     -- CTRL U SHIFT O CTRL U
Cancel Condensed    -- CTRL U SHIFT R CTRL U
Underline Mode      -- CTRL U FCTN R CTRL U SHIFT - CTRL U SHIFT A CTRL U
Cancel Underline    -- CTRL U FCTN R CTRL U SHIFT - CTRL U SHIFT @ CTRL U
Doublestrike Mode   -- CTRL U FCTN R CTRL U SHIFT G
Cancel Doublestrike -- CTRL U FCTN R CTRL U SHIFT H
Emphasized Mode     -- CTRL U FCTN R CTRL U SHIFT E
Cancel Emphasized   -- CTRL U FCTN R CTRL U SHIFT F
```

****

Try these with TI-WRITER "Editor" and your Gemini-10X Printer
Key them in before and after your text, as needed

LIST OF AVAILABLE PROGRAMS ON Poor Richards BBS for the TI-99/4A that are available for downloading through the use of XMODEM.

* TOLEDO BBS 300 - 1200 BAUD
* HOST SYSTEM: IBM PC
* PHONE 693-8282 300 BAUD
****

TEXIN.UTL
List of utility programs and such files
for Texas Instruments computers
as of Sat 11/30/85

...Name.....	..Size..	..Date...	Description
ANTIMATO	6528	85-11-11	Design and animate characters
CHAR_DEF	1920	85-10-12	Character Font generator program
TI-LOADE	2944	85-10-10	Disk menu utility
TRANSLAT	640	85-10-07	Translate ASCII files into BASIC pgms
BUG-OUTT	2048	85-10-07	Finds typos in mag pgms you typed in
TILOADER	2432	85-10-04	Auto-load menu on boot-up
PHONEDOC	2944	85-09-30	Doc file for phone book program PHONE

TEXIN.MUS
List of music programs and such files
for Texas Instruments computers
as of Sat 11/30/85

...Name.....	..Size..	..Date...	Description
GODFATHER	9344	85-11-17	Music program

.TEXIN.MISC
List of available graphics and misc. files
for Texas Instruments computers
as of Sat 11/30/85

...Name.....	..Size..	..Date...	Description
DONORS	1231	85-11-30	DONOR LIST
MORGANA.PIC	7040	85-11-30	Adults only, text file picture
BATGIRL.PIC	4992	85-11-30	Adults only, text file picture
NOVMINS	9984	85-11-28	November BBS usage list
BBS1000.TXT	75136	85-11-27	Numrers of over 1,000 BBS's nationwide
SWAPSHOP	2213	85-11-25	Odds and ends for sale from PR's
PHONEBOO	9344	85-11-17	New version of phonebook program
BBSLIST.TXT	1152	85-11-05	Local BBS Numbers
TICALC	7040	85-10-28	Spreadsheet program for the TI 99
VITALSIG	9088	85-10-28	Simulates circulatory system activity
YABBSLST.TXT	7788	85-10-14	Yet another BBS list
NANOPROS	8832	85-10-09	TI Machine language simulator
SEPMINS.TXT	9984	85-10-01	September BBS Usage summary
PHONE	7808	85-09-28	Little black book,0tracks your phone #'s
SPOCK	6810	85-09-19	Text picture of Spock
RAWSTARS.DOC	2816	85-09-03	Star table
YMODEM.DOC	44384	85-08-31	Discussion of XMODEM protocols
XMODPC	8192	85-08-04	More documentation on XMODEM protocol
BREAD.DOC	9344	85-06-26	Reprint from Readers Digest
CPROT.TXT	32512	85-05-18	Thread of messages from CS re: piracy
PIRATE2.DOC	12384	85-05-06	PIRATECO BBS listing sorted by number
BARBIE.TXT	6656	85-05-06	Nude figure (text file)
PIRATECO.TXT	43392	85-05-03	Software piracy conversation
SQUEEZ.DOC	14208	85-04-22	File compression techniques explained
ADFILE	6784	85-04-22	Advertising...
AREASORT	9728	85-04-21	Sorted list of area codes
XMODEM2.DOC	9984	85-04-19	Detailed explanation of XMODEM protocol
TRIVIA.DOC	10410	85-04-04	Trivia from the AD BOARD
FREQLIST.NUM	3456	85-03-11	Scanner Frequencies, Numeric sequence
FREQLIST.ALP	3456	85-03-11	Scanner frequencies, ALPHA Sequence
BEWARE	2688	85-03-03	Article about BBS Pirate
DOG	2560	85-02-23	Snoopy figure - ascii text file
PETTIPS.DOC	3656	85-02-23	Pet care info for winter
TOLBBS	2176	85-02-23	Numbers of some other Toledo BBS's's's
PCBBS.DOC	10624	85-02-23	Numbers of some other bulletin boards
ATLBBS	5504	85-02-23	List of Atlanta Bulletin Boards
XMODEM.DOC	2432	85-02-23	Brief description of XMODEM protocol
ARRESTED.DOC	3072	85-02-23	Article on BBS Legalities
PIRATES.DOC	12800	84-11-28	More BBS phone numbers

Information provided by
Roger Biddle, OH-MI-TI