WINNIPEG

Newsletter

October's Newsletter

The Winnipeg 99/4 User Group is a non-profit organization formed to meet the needs of Manitoba based Texas Instruments users. The content of this publication does not necessarily represent the view of the Winnipeg 99/4 User Group. This newsletter is one form of communication to keep Manitobans up on Texas Instruments Computers and its clones.

Next General Meeting - Date : November 7th, 1985
Time : 7:00 P.M.
Place: Winnipeg Centennial Library
2nd Floor, Assembly Room

Executive 1985:

President and Educational Co-ordinator:	Jim Bainard	889-4371
Treasurer:	Bill Quinn	837-7758
Newsletter Editor:	Mike Swiridenko	772-8565
Contributing Editor:	Paul Degner	586-6889
Inter-Group Representative and Newsletter Publisher:	Dave Wood	895-7067
Systems Co-Ordinator:	Sheldon Itscovich	633-0835
Public Domain Librarian:	Gordon Richards 822 Henderson Hwy.	668-4804
Module Librarian:	Peter Gould	889-5505
User Programs and Book Librarian:	Mike Swiridenko	772-8565

Mailing Address:

WINNIPEG 99/4 USERS GROUP
P.O.B. 1715
WINNIPEG, MANITOBA
CANADA, R3C 2Z6

BBS #: (204)-889-1432   SYSOP: Charles Carlson
SETTINGS: 300 baud, 7 bits, 1 stop bit, no parity
HOURS: 10:15 p.m. to 8:00 a.m. 7 days/week.

EDITORIAL COMMENTS:

Hello, again! I hope everyone is as well as can be. This month brings with it typical fall weather and anticipation of the cold winter months. It will soon be time to polish your winter boots and wax your cross-country skis. It seems that with the coming fall season we have the ripening of a new source of software, the 'Freeware Market'. If you are interested in finding out more I have included an article, taken from the 9T9 Users Group Newsletter, that tells you what FREEWARE is all about. I'd also like to say hello to all of the 99/4A User's Groups and Clubs that correspond with us. The number of letters we get keep increasing every month.

This months newsletter continues with the new columns started last issue. These columns include the user's tips, programmer's help file, and reviews section. Multi-Plan is reviewed, this issue, by Doug Hamilton (a budding assembly language programmer, and part time pastry chef). As usual Paul, in his column Quid pro Quo, provides his insight into the TI world.

That's about all! Hope you enjoy it. Read Miscellania to catch up on the latest club news.

If you have a review, user hints, or helpful programming tips, get them to me for the next newsletter. The deadline that I have set for submissions is one week before the date of the group's meeting. No matter how small you think your suggestion may be, it is quite possible others will benefit from it. Be an active member and participate! Thanks go out to Doug Hamilton, Doug Howe, Rick Lumsden, and all who have submitted items for this issue of our newsletter.

MISCELLANIA:

Miscellaneous news and reminders. Charles Carlson is now the SYSOP of a new BBS that is run on his TI. He expects to have an up/download section for TI programs, and a board dedicated to TI users. Charles calls his BBS the 'SMORGAS BOARD', and operates it between 10:15 p.m. and 8:00 a.m.. So if you have a modem and free time some night give his system a call (the BBS number is on the cover of this newsletter).

Hank Derkson has provided me with a letter, and catalogue, from Unisource Electronics that will interest all software/hardware purchasers in the club. Unisouce is offering a 4% credit to registered clubs for products purchased from them through a club representative. To register a club they are asking for a list of member names and addresses as well as the name of a person that will be the club's representative for all purchases. Unisource asks, also, that all orders be prepaid. I have passed the literature and catalogue on to Bill Quinn, our club's treasurer, as I believe that he may best represent our club. The 4% credit may be used to defer shipping costs, or to purchase products for the club, if we wish. This will be discussed at this months meeting.

We will be looking for businesses to place ads in next month's newsletter to subsidize the cost of publication. The price for a full page ad will be $10, half page $5, and appropriate rates for portions thereof.

There is some idle thought of holding a 'user written' software contest. As of yet the the details of categories, judging, and prizes have not been worked out.

Steve Zabarylo, a member of the Florida based National 99/4A User's Assistance Group, is looking for members to join this group. The group features software/hardware ordering, and send out a monthly newsletter. For every new member Steve signs up he receives a $10 software credit, which he is willing to give to our club so that we may purchase some items for our group. If enough members join we may be able to purchase some interesting stuff.

The club is still looking for new members and wishes to expand our membership in order to keep our club growing. As always, I am looking for new submissions to the user written program library. Any suggestions (donations?) for books, that could be added to the book library will be appreciated. Remember to check out the club libraries for new stuff. That's all for now.

READER RESPONSE:

This news article was submitted to me from Doug Howe and comes from September 12th's Globe and Mail.

SCIENTIST IS CHARGED WITH THEFT
REUTER NEWS AGENCY

DALLAS - Two former employees of Texas Instruments Inc. have been charged with stealing millions of dollars worth of top secret computer programs from the company.

A Dallas County prosecutor said felony charges that could lead to a penalty of from two to 10 years in jail have been filed against Thomas Schalk and Robert Leonard. Mr. Schalk, a computer scientist, is described by his new employer, Voice Control Systems Inc. of Dallas, as the chief inventor of a speech recognition system.

Both men left Dallas-based Texas Instruments to work for Voice Control, which is also developing computers capable of recognizing spoken words.

The indictments by a Dallas County grand jury allege that five Texas Instruments computer programs were stolen by Mr. Schalk on April 27, 1983. An additional 27 programs were alleged stolen last Feb. 18 by Mr. Leonard shortly before he left Texas Instruments.

REVIEWS:

This column presents reviews of materials that may be of interest to the user. The views expressed are the opinions of the reviewers, exclusively.

HARDWARE:

Charles Carlson will review his Myarc 128k card, and disk controller once he gets them back from the repair shop. Look for his review sometime in the next couple of months.

SOFTWARE:

The following review was submitted by Doug Hamilton and is of TI's Microsoft Multiplan. TI Multiplan needs 32K and at least one disk drive in order for you to be able to run it. A printer would also be beneficial, but is not necessary.

'The Joys of Microsoft Multiplan'

A review by Doug Hamilton.

My hands were shaking and the adrenalin was flowing as I unwrapped the package I just received from down south. This was it! The Big one. The Grandaddy of all the software--This was a SPREADSHEET. But would it live up to expectations or, even more importantly, would I be able to figure it out and make use of it.

It looked innocent enough, at first. A command module and a disk. Pretty much standard fare for TI software. The manual that came with it, however, told a grimmer tale. Used to as I am to the thin 50 page booklets that usually accompany such things, the 238 pages of the Multiplan manual was a terror to behold that brought back troubled memories of the Editor Assembler Manual of which can be said no more cryptic document is known to Man, Beast, or Computer Programmer.

Mercifully, the Multiplan manual is somewhat more tutorial in nature, and in fact should be given full marks for its detailed account of how to build a spreadsheet for the ficticious company, Spencer Ceramics. The beauty of this example is that it describes exactly what is to be done, why it is being done, and tells you the exact keys to press so that you, step by step, build the spreadsheet yourself.

By the time that you finish working through the Spencer Ceramics example (which, by the way, is 97 pages long) two things become very obvious. The first is that this is a marvelous piece of software capable of incredible flexibility, and secondly, notwithstanding all you have done so far, there still remains a large number of command options that you still have not used ie. you still have not scratched the surface of the computing power of this amazing tool.

Lets get into flexibility first. The worksheet is a grid of rows and columns, 63 columns wide and 255 rows long. Each little box in the grid is called a 'cell'. Within each cell you can enter one of three things: text, a number, or a formula. But, more than this is available. You can easily dictate the format of each cell by stipulating:

(1) The width of the cell.
(2) The way in which numbers appear eg. decimals, scientific, and dollars.
(3) The alignment eg. left, right, or center.

And all you PRK users will appreciate this-- You can alter the format of the sheet at any time as follows:

(1) Any of the above can be changed.
(2) You may insert or delete any number of rows or columns at any time or any place.

To cut down on unnecessary work some neat editting features have been included that can operate on whole groups of cells at one time.

(1) You can copy text, or data, or formulas from any row (or column) to any other area of the sheet.
(2) You can blank out cells in the same manner.

As one works through Spencer Ceramics, greater fondness for the above mentioned features will occur. Suffice it to say that once the admittedly formidable number of editing commands have been mastered there should be very little unnecessary repetition of entries happening.

Ahh! Now, the truly amazing part. The calculating features. Consider some of the following tender morsels.

(1) Full range of mathematical formula including arithmetic, log, and trig functions.
(2) Built in formulae to find the max, min, Net present value, average, and standard deviation.
(3) Full range of Logical functions. ie. AND, NOT, etc.
(4) The ability to assign a name to a whole row/col, eg. name col 3 SALES. To sum all of the entries of column SALES, all you have to do is place the formula SUM(SALES) in a cell located elsewhere.
(5) Lookup function that can be used to find data in a table - an excellent tool that, as is shown in the manual, could be used for finding the tax payable from tax tables.
(6) Sorting option which distinguishes Numbers, Text, Logical and error values, and Blank cells.
(7) An iteration option that can repeat a calculation using results from the previous calculation- an example from Spencer Ceramics is included. Other applications could be binary search models, or interest tables, or ???? how good is your imagination ????

Feeling a little overwhelmed -- that's good! I would be crushed if you weren't , as I have been studying this for a while and am still a novice. Its clear that full application of the calculating power is as likely to be limited by the user's ability as it is by the shortfalls of the program.

Anybody like icing on their cake?? Well the Microsoft people sure do. Consider this, we know that the size of the worksheet is 63x255 which, lets face it, is not in the mega category, and besides with a 32K configuration only so much can be done. The Microsoft people, however, have made an admirable attempt to overcome this difficulty by a process that they call LINKING. What is LINKING? Suppose you have two worksheets the first of which is called ASSETS, and the second of which is called SALES. Now the worksheet SALES contains the detailed breakdown of all the sales done on a daily basis, and one cell on the worksheet has been named Sales. On the worksheet ASSETS there is also a cell named Sales. By linking the cell on SALES with the cell on ASSETS, the cell on ASSETS will be automatically updated whenever the corresponding cell on SALES is changed. For example, suppose that you have saved both SALES and ASSETS on a disk. You then load up SALES, enter the figures for today's sales, and save the worksheet to disk. After the worksheet is saved you will notice that the red light on your drive is still blinking- what is it doing?? It is searching the disk for any other worksheet that is linked to SALES, and when it finds ASSETS will automatically update it!!!!!! Think about that for a moment. With linked sheets you can, with careful planning, have a sheet that is literally the size of an entire disk.

Other nice features include:

(1) Printing options that allow you to print only those parts of the sheet that you want. You can set margins. Also, you have the option of printing the row/col numbers, and the option of having the formula that is in the cell printed.
(2) Windows- these allow you to view specific parts of your spreadsheet continully while viewing other sections of it. Sort of like glueing parts of the spreadsheet together so that different areas, of the spreadsheet, can be viewed at the same time. You can even put a border around a window.
(3) You can shut off the calculating function when entering data, so that you can enter data faster, and let the program do all the calculating after the data has been entered.
(4) You can lock up cells so that the formula in them cannot be altered accidentally.

(5) A goto command allows you to get get to a specific cell quickly. Along with this is a Home function that takes you to the upper left corner, and a function that takes you to the lower right.
(6) An overlay comes with the package to aid in keying.
(7) All commands are displayed on the screen and can be accessed with one letter.
(8) The data or formula of a cell is (or can be) displayed on the screen.
(9) The percentage of space left on the sheet is shown on the screen.
(10) There is a help command that fully explains the desired option.

Anyone like cherry on their cake?? I sure do! In this case the cherry is a particularly useful one, that being, that you can save your spreadsheet in such a way that the TI writer can call it up as it would any other saved letter. You can then add text to your work; thereby, producing very professional looking reports. For those of you that like bar graphs this option is also available, and prints as many asterisks as the value in the cell.

There is available an update, to the Multiplan, that speeds up the process somewhat and gives an autorepeat feature to the cursor.

The applications of this program are endless and I would suggest that anyone who has both the TI Writer and the Multiplan, and does not want get involved in programming, will find most of their sofware needs met. Certainly all the standard programs that do banking or personal financial calculations would be taken care of, and specialized calculations that might be required by a tax consultant, building contractor to do estimates, or most other calculations of costs could be handled. Like anything it will take practice, and some thought, before you are able to solve your particular needs. If anyone out there has a spreadsheet and needs help to solve a specific problem I would enjoy trying to solve it, but cannot guarantee a solution.

BOOKS:

Purchase a book, about the TI, lately and feel good, bad, or indifferent about it. This space is available for your comments.

HELPFUL HINTS AND TIPS!
(FOR THE USERS, BY THE USERS!)

This column features tips brought to my attention from members of this group, other user group's newsletters, and various other sources. **WARNING:** These hints and tips are to be used at your own risk!

PRINTERS:

If you are interested in multicolor printouts you can buy ribbons, for your printer, of different colors and set up your program to let you swap ribbons for the desired effects. Not quite as good as a plotter but a heck of a lot cheaper.

MINI-MEMORY:

If you have a disk system I have the Line-by-Line Assembler on disk. If you want a copy contact me for one. I also have the lines demo program available as a relocatable object file.

EDITOR/ASSEMBLER MODULE:

If you only have a single drive and with to do assembly language programming it is a good idea to copy the files: EDIT1, ASSM1, and ASM2, onto the disk that you will be using for your source and object files. This will save you swapping the Editior Assembler disk with your assembly language program disk. If you have two disk drives the Editor Assembler disk must me placed in drive #1, as this is the drive that the module looks to for its files.

HARDWARE:

Frequently one comes across the terms Third Generation computer or, with ever increasing frequency, Fifth Generation computers. These terms refer to specific stages in the development of computer hardware. The First Generation computer was constucted using vacuum tubes. Second generation computers saw vast improvement in space and energy savings with the development of the transistor. Third generation computers developed further with the techical ability to place many circuits on a single microchip. Fourth generation computers (sometimes called Super computers) depend upon VLSI (Very Large Scale Integration) of many microchip circuits for their speed. Fifth generation computers are at the head of current computer development, and look to design of faster processors and better memory schemes. Fifth generation computers are hoped to be the start of the 'smart' computer era. With every advance in hardware development has come corresponding advances in software development. Software may be termed as to be First Generation, Second Generation, etc. Software has developed along different lines, and are another topic all together.[ed.]

TERMINAL-EMULATOR II:

There are several public domain terminal programs avaialble in our public domain library. Contact Gordon Richards.

MULTI-PLAN:

Turn to the reviews column for a review of Multiplan by Doug Hamilton.

TI-WRITER

Have you ever wondered how you could get the TI-WRITER to print 132 characters per line? The following tip from Paul A. Meadows, of the TI-Nova Scotia Club, tells you how. This information was taken from TINS' September newsletter and has been edited for space reasons.

One or two precautions are warrented, first. If you are using tabulated information you will have to exercise extreme care, the FILL and ADJUST commands are essential to this process. Some experimentation with different form types might be a good idea, to get the feel of the technique.

Now to the steps...

1. Prepare your file as you normally would with TI-Writer.
2. When you have finished typing in your text, go to line 0001 and enter the margins and commands, for example ".LM 10;RM 120;IN +5;FI;AD". This will set the left margin at a good starting point and the right margin will balance on the opposite side. The indent command, of course, is up to you. The Fill and Adjust are necessary. Watch out for any tab'ed information!
3. Now, instead of SF, for "save file", you use the PF, or "print file", command preceded by the letter "C", which strips all control codes from the file. Along with the normal control codes, the TI-Writer tab line, that was applied to the very last line of the file, is stripped. Thus taking away the mandatory 80 column indices.
4. Once the program has been "printed" to the disk using the above format, "C DSK1.FILE", you can exit the editor section and select the formatter.
5. Select formatter command, option 2, and follow the prompts to print out the file.

Since the file no longer has the TI-Writer tab indices to look up, it will execute the internal margin and tab instructions.

There you have it! Simple isn't it. (Thanks, Paul. [Ed.])

EXTENDED BASIC

From the 9T9 newsletter comes this tip, by Graham Hilton England.

HAVING TWO PROGRAMS IN MEMORY AT ONCE. you will need the 32K RAM and Extended Basic.

1. Select Extended Basic.
2. OLD the first program. It will go into 32K.
3. CALL PEEK(-31952,A,B,C,D).
4. PRINT A;B;C;D. Note down these line number table pointers.
5. Make the 32K invisible with CALL LOAD(-31868,0,0) which disables the 32K memory.
6. OLD the second program. It will go into VDP RAM.
7. When finnished running, NEW, and restore the 32K RAM using CALL LOAD(-31952,255,231).
8. Reload line number table pointers by CALL LOAD(-31952,A,B,C,D).
9. Now run the program in 32K.
10. You can even QUIT or BYE or even switch off the console without losing the program in the 32K.

TI PROTECTION SCHEMES:

The following was compiled, and submitted by Rick Lumsden. Rick collected these gems from various publications, supporting the 99/4A.

Propriety disk (DMII-FCTN X)
Sector 0 address 0010   2028=Unprotected
5028=Protected
Write Protection (DMII-Protect)
sector 2-33 address 00C

At the File Descriptor Block in sectors 2-33 find the file you are looking for with search option. At address 00C0 there is a value that changes depending on the type of file.

1.Program->01
2.Dis/Fix->00
3.Int/Fix->02
4.Dis/Var->80
5.Dis/Int->82

A value of >08 is added to this to install write protection. Remove by subtracting >08 or use DMII.

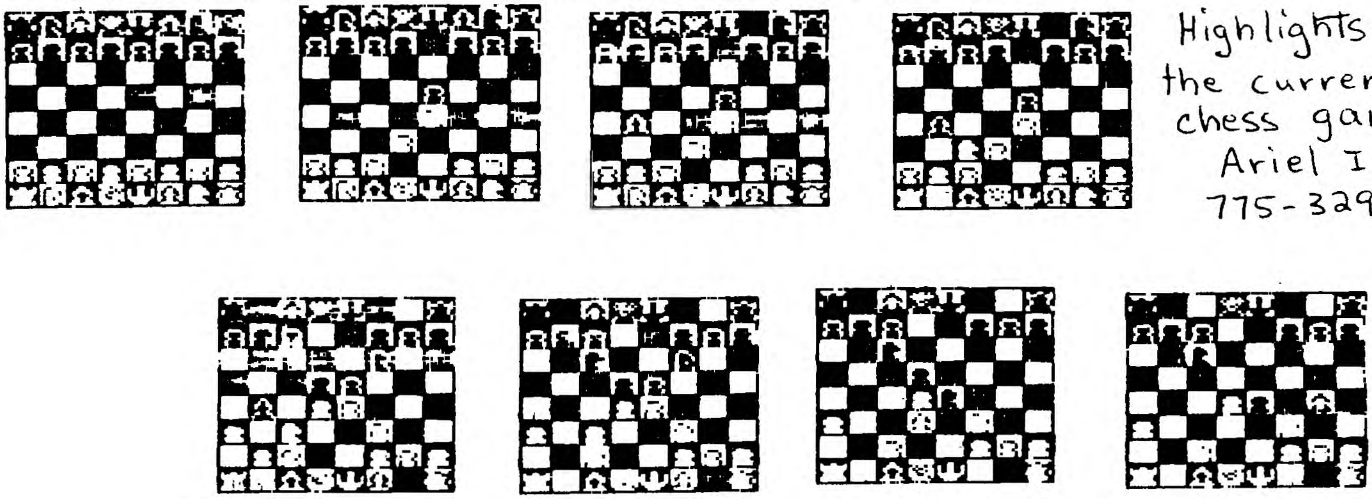
Extended Basic Protection (SAVE DSKx.xxxxx,PROTECTED)

Address 0000 First sector of program. Subtract the value of address 0000 from >FFFF and add 1 then insert into address 0000.

(eg. protected-> 0000=FFFD FFFF-FFFD=0002 0002+0001=0003 <-unprotected)

This protection can also be removed from Extended Basic as follows.

1. Load program.
2. Call Init :: Call Peek(-31868,A,B,C,D) :: Call Peek(-31952,E,F,G,H)
3. Write down the values obtained.
4. NEW
5. Call Load(-31868,A,B,C,D) :: Call Load(-31952,E,F,G,H)
6. A,B,C,D,E,F,G,H are equal to the values obtained with the Call Peek command.



Highlights of the current chess game on Ariel I BBS. 775-3298

FREEWARE By Danny Michael

Reprinted from the August 9T9 User's group newsletter, who reprinted it from The Computer Bridge Newsletter of the St. Louis 99'ers. May/June 1985. Vol 4 #5/6.

Over the last year or so, a new buzzword has been gaining popularity in the TI community. The word is FREEWARE and it's been bringing new and useful software to TI users at prices that are agreeable to everyone. But as always with new concepts, there has been some confusion among some users as to how the Freeware system works, and a misunderstanding among others as to the responsibilities of the user who wants to benefit from the Freeware marketing scheme. But first, just what is FREEWARE?

FREEWARE is a marketing technique in which a software author offers his program(s) to the public at no charge. Contrary to the usual 'commercial' copyright notices that inform you that you are breaking the law if you copy the program, FREEWARE programs actually contain a notice encouraging you to make copies of the programs for your friends. Although most FREEWARE works are copyrighted you are in effect given a liscence to distribute the program to others, providing that you do this at no charge. In other words you can't sell the programs, but you can give them away to anyone.

Now that you know what FREEWARE is, don't you wonder why anyone would want to spend their time developing software just to give it away? Well, that's where some of the misconceptions are, and the thing that throws some people off is the name itself... FREEware. FREE - something for nothing, right? RIGHT! WRONG! Its both right and wrong. You'll notice that I called FREEWARE a marketing technique. FREEWARE programs contain a notice that informs you that if you like the program, and think its worth paying for, to send your payment to the author. Its sort of like test driving a car before you buy it. The difference is that if you decide not to pay for the program, you don't send it back.

The price you pay, if decide to do so, depends upon the program. Some authors suggest a certain price (generally $10), others tell you to send what you think the program is worth, up to a maximum limit (again, usually $10), while others do not put a limit on the amount. The main consideration, of course, is the value of the program to you. But you must also consider that the author is providing his time and talent to produce software for your orphaned computer with no guarantee of any compensation for his work.

What kind of people do this? Authors of FREEWARE programs come from many professions and backgrounds. The reasons for the decision to market a program through the Freeware concept rather than the more traditional commercial ways are just as varied as the people. Some people simply want to share their work with others, and the money they receive is just icing on the cake. Others write programs with the intent of selling them commercially, only to find that there's not much money to be made in the TI software market place, due mainly (and sadly) to the passing around of commercial software within users groups. The Freeware route gives those persons receiving copies the opportunity to pay for the program if they'd like to.

For the most part, FREEWARE programs are of 'commercial' quality. Many are of higher quality than the commercial programs of the same kind. The only thing you will probably will not get with FREEWARE is the printed instruction manual. Most documentation is provided from within the program, and any written instructions are usually provided as a text file, on the disk, which you must print for yourself.

Now that you understand the concept of FREEWARE, you should be aware of your responsibilities as a user. If you receive a FREEWARE program from a friend or a users group, you have two decisions. First and foremost is whether or not to pay for it. This decision should be made with respect for the considerations outlined above. The easiest decision to make is whether or not to give copies to others. Remember, you are a vital part of the FREEWARE wheel. The more people that are exposed to an authors programs, the more money he is likely to receive for his efforts., which makes it more likely that he will continue to make quality software available for users.

If you see an announcement for a Freeware program in a newsletter, or magazine, and would like to try it out, your repsonsibilites are greater. Again, the word FREE throw some folks off. As you know there is no such thing as a free lunch. You can't just write or call a FREEWARE author and ask him to send the program. These are not big businesses with big budgets. You are expected to provide the recording media (in most cases a floppy disk), and the cost of returning the media to you. That means that the program is not 100% free to you. You should always send an initialized disk, a mailer that is in good shape, a return address label, and sufficient postage to mail the disk back to you.

Some Freeware authors will provide all this for you for a small fee, but remember, that the fee only covers the media, mailer, and postage, and should not be considered as payment for the program.

Most Freeware announcements will tell you what is expected by the particular author. If you're not sure, go a little overboard. Send an initialized disk, stamped, self addressed disk mailer, and a letter stating your request and your willingness to provide anything else necessary to receive the program. By all means, do send a note or letter stating your request. NEVER just send a disk. Some authrs have more than one progam available, and its hard to read your mind on the other side of the country!

That brings up a point. Unless the announcement specifically states otherwise, send a seperate disk for each program requested. Again, in most cases the announcement will fill you in on the details. If it mentions a 'disk full of programs' then one disk will probably do, but if it tells of two or more programs seperately then its good practice to send a disk for each program. Remember to always send initialized disks. This cuts down on the time for the author to return a copy to you. And be patient. For the most part, FREEWARE authors have regular jobs, and their time, like everyone's, is at a premium.

After you have received your copy of the program(s), your responsibilies are the same as outlined for persons who got the programs from a friend. If you have questions, comments, etc. about the program and decide to contact the author just remember to make it easy for him to accommodate you. If the documentation with the program lists the author's phone number, then most likely he will be willing to talk to you if you call. Just be sure to call at a reasonable hour, usually not after 9 PM in the author's time zone. If you write, be sure to send a legal size SASE for the reply. Don't expect the guy to bend over backwards to answer your questions if he has to provide the envelope and postage to send the answer to you.

Hopefully, you now have a better understanding of the FREEWARE marketing system. Below is a list of persons who have Freeware available, with a short description of their program(s). This listing is by no means a complete listing of all Freeware programs, just the ones I know of. The doallar amount specified is the author's suggested payment price for the program. All the authors listed require an initialized disk, addressed disk mailer, and return postage as the minimum for returning their program to you. Some offer to provide these items for a small fee. Those are noted.

Steven Lawles, 2514 Maple Ave., Wilmington, DE 19808.
** MASSCOPY ** ($10)

A disk copy utility. Will utilize the FOUNDATION 128K card if you have it. Also will make 2 copies if you have 3 disk drives.

Emile Verkerk, 29 Ingleside Drive, Downsview, Ontario, Canada M3K 1V2.
** DISK INVENTORY MANAGEMENT SYSTEM ** ($10 + disk) Extended Basic.

Manual system to keep track of 1 to 1,000 disks, instant lookup, with 8 pages of documentation.

Clint Pulley, 38 Townsend Ave., Burlington, Ontario, Canada L7T 1Y6.
** SUPER COPY ** ($10)
** 9900 BREAK-THRU ** ($10).

Super Copy is a disk copy program for E/A or Mini-memory. Comes with source code. Break-Thru is a TI version of the popular Breakout arcade game. Clint cannot use U.S. stamps as postage. Send him $1.00 to cover return postage costs.

Gary Cox, 3174 Melbourne, Memphis, TN 38127.
** WEATHER FORCASTER ** ($5) in Extended Basic.

(Good stuff, eh! I hope to bring you more FREEWARE addresses the next newsletter. Micropendium puts out a list every second month. [Ed.])

PROGRAMMING HELP FILE:

The purpose of this column is to present, to the user, techniques that will be useful in the writing of programs for the TI-99/4A home computer. As not all readers will have the same programming skills I will present the material at a fundamental level but in a way that tries to be stimulating to the more experienced. Not everyone has knowledge about assembly language and I hope that what is presented here will get some of those people started with it. I hope that there is something, in what follows, for everyone. If you can provide some prgramming insight that might be useful to someone, please, feel free to pass it on to me, and I'll get it into the next newsletter.

BASIC/EX-BASIC:

The DEF statement is useful when you need a single line operation that is repeated in several places in your program. The function defined by a DEF statement acts as a user written function call. TI basic has several builtin functions, of the most familiar are SIN, COS, TAN, SQR, LEN, ASC, VAL, CHR$, MIN, MAX, and SGN. You can use MAX and MIN to limit a number to a certain range of values, as in the checking of the screen boundaries for graphics programs.
For example: COL=MIN(32,MAX(1,COL+X/4)) will limit the value of COL to a number between 1 and 32, given the previous value of COL, and the value X for horizontal direction of joystick input. MAX, and MIN are both examples of 'two parameter' functions. DEFined functions can only have one parameter; however, they may refer to variables that exist in the rest of the program (called global variables). A function that is not in either of the TI basics is the Modulus function. The modulus determines the remainder of a number with respect specific divisors. Such a function is useful for HEX->DEC, and other such number conversions. It looks like this when the divisor is 16:
DEF MOD(X)=X-16*INT(X/16).
Note that X is a variable that has no meaning to the rest of your program as is used to contain the value of the number that is given to this function. When the function returns it will not change the value of any variable called X that may exist in the rest of your program. A call to the function MOD will look like this:
REMNDER=MOD(243).

For more information on any of the TI functions discussed here consult your Extended Basic reference manual.

ASSEMBLY:

The following program tips were inspired by COMPUTES! 'Beginner's Guide to Assembly Language on the TI-99/4A' by Peter M.L. Lottrup. This book is on order and will soon be in the club's book library. As you will recall from last month's newsletter I had a simple number guess program in this section. The program looked long, but that is because you are so used to Extended Basic programs. To program in assembly you must define every step that the computer must perform. This is a tedious process requiring careful thought, and planning. It is not difficult to do, it just requires the desire to sit through it. The rewards of assembly language programming; however, depend on how much you are willing to program in it. Enough talk. What follows are a couple of segments of code that explain parts of last month's program.

To display a character to the screen is as simple as this:

```
LI   R0,@SCREENPOS   <- LOAD REGISTER 0 WITH A NUMBER BETWEEN 0 AND 273.
USE THIS CONVERSION FROM ROW AND COLUMN TO FIND THE APPROPRIATE LOCATION.
SCREENPOS = (COL-1)+(ROW-1)*32
LI   R1,@CHARCODE    <- THIS IS THE SINGLE BYTE HEXADECIMAL CODE OF A CHARACTER THAT YOU WANT TO DISPLAY.
BLWP @VSBW           <- CALL THE CONSOLE ROM UTILITY TO WRITE THE CHARACTER.
THIS IS THE 'SINGLE BYTE WRITE TO VIDEO' ROUTINE.
```

To display a line of text to the screen is almost as simple:

```
LI   R0,@SCREENPOS   <- SAME AS ABOVE.
LI   R1,@MESSAGE     <- LOAD R1 WITH THE ADDRESS OF THE TEXT STRING.
LI   R2,@MSGLEN      <- LOAD R2 WITH THE LENGTH OF THE MESSAGE. IE. NUMBER OF CHARACTERS.
```

```
BLWP @VMBW          <- CALL THE 'MULTIPLE BYTE WRITE TO VIDEO' CONSOLE ROUTINE.
```

NOTE: The console routines outlined above look to registers R0, R1, (and R2 in the second case) of your workspace registers, for information that they need.

If you're with me this far I can tell you a little about how assembly language operates.

Your computer operates on memory locations that consist of bits and bytes. A byte is made up of 8 bits. A bit is simply a single switch that can hold the value of one or zero (on or off). To work with bits you can use binary arithmetic and are thus able to define numbers that are made up of 8 bits. Binary arithmetic unlike decimal arithmetic has only two allowable digits, 0 and 1. Thus the binary string '00001010' will represent a decimal number of some value. In decimal arithmetic each digit, going left from the decimal point, takes a value of increasing powers of 10, eg. 1, 10, 100, 1000, and so on. Similarly, a binary number, because it has only two numbers it may use per digit (0 or 1), will take on increasing powers of 2. eg. 1, 2, 4, 8, 16, and so on. You can now see that the decimal value of '00001010' is 8 + 2 or ten. An 8 bit number can take on values from 0 to 255. A 16 bit number, called a 'word' of memory, can take on values from 0 to 65,535.

You may recognize the value of 65,535 as the number of bytes that are in 64K of TI's memory. This is because the TI has 16 address lines that are used to select individual bytes from its memory. Each line may be on or off, ie. 0, or 1.

Once you know why the memory is organized the way it is, you can better understand how assembly language works with it. This is just the beginning of learning to program using assembly language. I will continue this discussion next month.

FORTH:

Alas the long awaited Forth column. The following discussion is of input and output words from the Forth language. I will use TI forth. For corresponding words in your favorite Forth consult the appropriate manual.

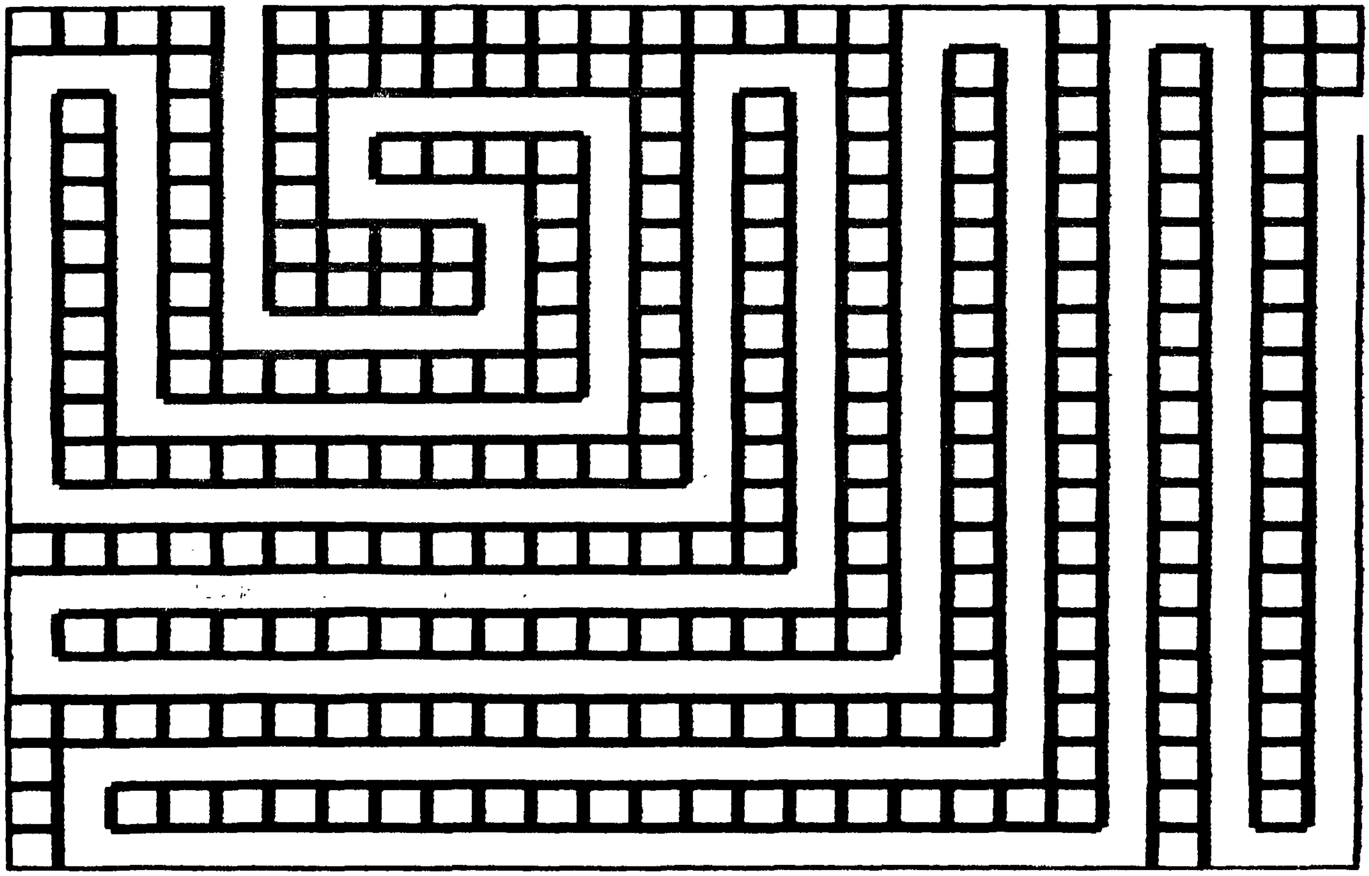
This month I will discuss Forth input and output words. I will start by mentioning the output words.

When using Forth you will want to be able to display things from the stack. To do so you can use '.'. This will pop and display the top element of the Forth stack. To print a character string you can use '." xxxxx"', where 'xxxx' is your text item. Notice that there is one space directly after the '." '. That space is necessary for Forth to identify the start of that command. The print string ('." xxx"') is useful when you want to print messages from within your defined words. Another word is use to display a single ascii character to the screen. It is 'EMIT'. Simply specify the ascii code of the character you wish displayed before the word 'EMIT', eg. 78 EMIT. The character is then printed onto the screen. Another useful word is 'CR'. It simply performs a carriage return, and is used to after printing a line. 'n SPACES' can be use to print 'n' spaces. 'SPACE' will print a single space. 'CLS' is used to clear the screen and is found in the -SYNONYMS package. 'n SCREEN' of the -GRAPH package, alows you to change the screen color to the color (n) that you want. '.S', located in the -SYNONYMS package allows you to display the contents of the entire stack, without poping any of the items from the stack.

After mastering output you may want to try a bit of input. It is usually a good thing to know if you have a word that is to interact with some user. Here are the two Forth words that will do the job.

'KEY' acts much like a INPUT from basic, except that it only takes one key press for input, and leaves its ascii value on the stack. This word will wait for a key press. '?KEY', on the other hand, acts like the CALL KEY of basic, and leaves the ascii code, of a key pressed, on the stack. If a key was not pressed a 0 is left on the stack.

Well there you have it. Input and output words in TI Forth. Try them out to see what you can do with them.



The Winnipeg 99/4 Users Group

Quid pro Quo

By Paul Degner

So this is October when we venture into our garages to tune up our snowblowers or in my case the trusty old snow shovel. For those of you who con the wives into doing this then you will have some time on your hands to play around with the computer. You'll need something to make the computer play with so why don't you consider the following excellent ideas.

Briefs:

- The September issue of the Millers Graphics Newsletter, The Smart Programmer, really got my bank book doing hoola hoops. I won't say anything more than to take a look at the accompaning page which has been photocopied from their newsletter.
- You may have noticed in the September issue of Compute in their classified section under hardware that Tim Northrup is advertising a 32K internal upgrade. Further light has been shed upon this from TRS Systems. Miller Graphics quotes 'There is a New 32K 16 BIT RAM console modification for our 4As. This mod will allow all of your Assembly programs to run as if they are in Scratch Pad RAM - very fast! By making all of the 32K Memory Expansion 16 BIT it really opens up the 9900 micro and allows it to run without WAIT states in between each and every byte! We've been told that the speed increase is any where from 20 to 50% FASTER. To have one of your consoles modified just send it to TRS Systems: 17885 Mount Elliot: Detroit, MI 48212; (313)366-9088 along with a check or money order in U.S. funds for $125.00. This upgrade has a 1 year warranty.'
- Still waiting for Myarc's Extended Basic module to come to Winnipeg? Well Miller Graphics points out that a well know software manufacturer in West Germany has developed a new module that is an exact duplicate of TIs but has alot of new enhancements in CALLs. These were developed by Heiner Martin. They include: GSAVE, GLOAD, BHCOPY, VPEEK, GPEEK, VPOKE, ALLSET, WAIT, MOVE, MLOAD, MSAVE, BYE, NEW, RESTORE, QUITOF, QUITON, SPRON, SPROF, FIND and APESOFT. CALL APESOFT loads the Hi-Resolution graphic LINKs into Low Memory Expansion. They allow you to draw and display line and graphs on a pixel by pixel basis. It does not use bit map mode but instead uses a technique similar to TI Logo with character redefinition. The module should cost around 120 U.S. Dollars. For more information contact MECHATRONIC GmbH; Dresdener StraBe 21; 7032 Sindelfingen; West Germany.
- Does it seem that the software market is drying up for our pal the 4A. Well not so, look again. A publication that I subscribe to has brought out a few new programs to titillate the errogenous zones. Night Mission is a book and cassette combination. The cassette contains the award winning fast action 5 screen heliocopter rescue game of Night Mission which is written in Extended Basic. The book, which picks up where the Smart Programming Guide for Sprites left off, fully documents each line of the program flow. It is complete with screen diagrams, character lists with diagrams, and variable lists. Another chapter in the book explains the use of the powerful AND function, through example listings, to help speed up your programs and to save bytes. There is also a section on CALL PEEK and CALL LOAD that lists many various locations in the computer and documents their use. Also the game is available on disk. Price $19.95.
- Explorer is a diagnostic utility to explore the memories of your 4A. It is an excellent program as I've been spending long nights probing, proding, fondling, and caressing various memory addresses in the 4A and these sleepless nights are all to the thanks to Charles Carlson for bringing this program over. It also comes with an excellent manual detailing all of Explorer's functions as well as various memory maps. Price $24.95
- Advanced Diagnostics you may think is just another hum drum memory prober but boy are you in for a surprise! I have ordered this piece of software so a review should be in the next newsletter but I can give you a glimpse of what it does. It does macros, beep, change colors, check disk, command file, check memory, convert number, copy read, copy write, disk directory, edit sector, format disk, find file, head step, motor speed, output device, output width, pause, read track, seek track, select drive, time delay, use dsr, write sector, and write track. If that is not all it comes with 40 pages of documentation that also includes an introduction into the theory of operation of floppy disk controllers and complete memory maps of the important sectors on a diskette. Price $19.95
- Now I bet you are wondering who produces all of these programs well I guess I can breathe a word about them. Miller Graphics at 1475 W. Cypress Ave; San Dimas, CA 91773; (714)599-1431. Shipping charge is $2. All funds payable by Money Order in U.S. Currency.
- Getting tired of the sweat stained Joystick? Looking for a new way to draw images on the screen? Well I received the September issue of TINS and found a article on how to design a light pen for the 4A. The nice thing about it is the price. You are looking at one nine pin D-plug, six lengths of two conductor wire, eight CDS photocells, and eight junk flair pens (or anything big enough to accomodate the photocells.) Now once you have all the necessary materials get out the pens and cut off the tips of the pens as well as punch out the ends of the caps. Feed the wire through the bottom of the pen out the tip. Solder the two wire to the leads of the CDS cell, and place the cell into the cap and put it on the old pen body. Tape the wire around the pen to help prevent it from pulling out by accident. Grab the D-Plug and hard wire the leads for the following: pen up one (7 common and 3 up), pen left one (7 common and 5 left), pen down one (7 common and 8 down), pen right one (7 common and 9 right), pen up left diagonal one (7 common and 3 up 5 left), pen up right diagonal one (7 common and 3 up 9 right), pen down right one (7 common and 8 down 9 right), and pen down left one 8 down 5 left). A person could simplify this by using eight switches and one CDS photocell which should save you even more money. The software to use this isn't to hard to program up so I will leave it to you to figure out.
- Don't forget to check into our public domain library because we usually forget to mention new programs that have come to us and are just popped into the library. We have many excellent programs so give Gordon Richards a call!
- A update on groups to whom we communicate with has brought us the Jackson-Metro User Group of Jackson MS, SHER-TI of Sherbrooke Quebec, and the Sudbury 99er User Group. A hardy welcome to all and hope a long friendship will exist between the groups.
- That's all for now! Remember Yak farming is inexpensive!

GRAM KRACKER™

Boy, do we have something NEW for you! We have completed the prototype and testing of some new hardware that Millers Graphics is going to produce and it should be ready to ship by November 15. This unit plugs into the module port and it contains 48K of RAM and GRAM. That's right GRAM, the RAM complement to GROM. We call this unit the GRAM KRACKER^{tm} and we are VERY pleased with the versatility and power of this little unit. You will be able to plug ANY module into the GRAM KRACKER^{tm} and SAVE its contents to Disk or Cassette. Once saved, they can be loaded into the battery backed up GRAM KRACKER^{tm} and run.

It is very simple to operate. On the front panel there are 4 switches and a Reset button. The switches control the loader, Ram banks and write protect, and Enable or Disable of GRAMs 0, 1 & 2. The GRAM KRACKER^{tm} also contains sockets on its circuit board to allow you to expand its total programmable memory to 80K of RAM and GRAM. This gives you 16K of bank switched Cartridge ROM/RAM and 64K of Console and Cartridge GROM/GRAM. This means you can now modify the contents of GRAM 0, the system monitor, GRAMs 1&2, TI Basic, and GRAMs 3-7, in the cartridge. You can also modify the contents of Cartridge RAM >6000->7FFF (2 banks) to suit your needs. So now you can change the Title Screen, change menus, have true ascenders and descenders for your character sets. You can change the default screen and character colors for any module. You can change the default printer configuration for your modules. You can add many new CALLs to TI Basic or TI Extended Basic. You can override TI Basic and put something else in its place. This would allow you to have a menu with a couple of different modules to select from.

With a full 80K GRAM KRACKER^{tm} you can have a menu that includes Editor/Assembler, TI Writer, Extended Basic and 1 more single Grom chip cartridge like Adventure. Once you have the GRAM KRACKER^{tm} configured the way you want, and the modifications made to the software in it, you can save it all out to Disk or Cassette. Since the GRAM KRACKER^{tm} is battery backed up, you DO NOT have to reload it each time you turn on your computer!

Now you can take advantage of the powerful GPL Language that is built into our 4As. Through Assembly Language GPL Link you can execute GPL routines that YOU have set up. There are many routines in the interpreter that perform common tasks quite nicely. For example; the GPL code 07 20 00 will clear a 32 column screen and return to your Assembly program. This is the opcode for ALL >20 RETURN. Once you've loaded this opcode into GRAM just execute a BLWP @GPLLNK DATA >xxxx, where xxxx = the GRAM address where the opcode is. Since GRAM is auto-incrementing memory,like GROM, it has an added bonus in that it loves to store and retrieve DATA. So with only the Editor/Assembler module loaded into the GRAM KRACKER^{tm} you still have 32K of GRAM DATA storage AND 16K of bank switched cartridge RAM plus ALL of memory expansion to use. So that's 80K of Program and DATA area and we haven't even touched VDP Ram yet! With an 80K GRAM KRACKER^{tm} and Memory Expansion we now have 64K of GRAM, 16K of Cartridge RAM, 32K of Memory Expansion RAM and 16K of VDP RAM for a total of 128K of RAM & GRAM. Oh those clever folks at TI, if they would have only opened this up to us sooner! Who says the TI 99/4A isn't a programmers machine!

The GRAM KRACKER^{tm} has an additional 8K of programming built in to it for the SAVE, LOAD and EDIT routines, which are enabled and brought up on the menu with one of the switches. It also comes with a floppy diskette that contains some modifications and enhancements for Extended Basic. The GRAM KRACKER^{tm} will be produced in limited quantities at first and the quality will be guaranteed by us. We are not going to ramp up production into high gear only to have to file chapter 11 because of miss management or over zealous production. We want to build a Reliable, Long Lasting, Quality product. After this project is on the road we have many more items planned. And you thought the 99/4A was an orphan? Little Orphan Annie is more like it!

By the time you read this the GRAM KRACKER^{tm} will be ready to start production and as soon as it is ready to ship we will mail out brochures. The price will be in the 150 to 200 range, we're finalizing parts prices with the suppliers now. Finally the secret world of GPL is open to us to play, modify and program in, and is it a lot of FUN!

MEMBER WRITTEN PROGRAM LIBRARY REGULATIONS

1) The W99/4UG Member Written Program Library shall consist of owner written and/or translated programs. Programs copied from any magazine will not be accepted because they are to be considered as public domain software.

2) The W99/4UG Member Written Program Library will not be responsible for any bugs in any program submitted, nor shall the W99/4UG Member Written Program Library make changes to any program. Corrections will be the responsibility of the owner and it will be the responsibilty of that person to send a corrected version of that program to the librarian.

3) The W99/4UG Member Written Program Library shall charge:

$2.00 per program (goes to author)

$1.00 for cassette or provide replacement cassette

$2.00 for diskette or provide replacement diskette

4) Instead of paying for a program a member may use the software exchange program. Any member that submits a program to the W99/4UG Member Written Program Library will receive three programs in exchange.

5) The W99/4UG Member Written Program Library is open to any member of the Winnipeg 99/4 User Group.

6) Any member found pirating another member's program will be asked to leave the group.

7) The Winnipeg 99/4 User Group is not responsible for the content of any program in its library.

PUBLIC DOMAIN LIBRARY REGULATIONS

1) The W99/4UG Public Domain Library shall consist of user written and/or magazine published programs who have publically stated the release of that program as public domain material.

2) The W99/4UG Public Domain Library will not be responsible for any bugs in any program submitted, nor shall the W99/4UG Public Domain Library correct or update any program. Corrections and/or updates of the program will accepted from any member of the group as long as the program contains the original programmers name.

3) The W99/4UG Public Domain Library is open to any member of the Winnipeg 99/4 User Group.

4) Any member found placing commercial or unauthorized software within the public domain library will be asked to leave the group.

5) The Winnipeg 99/4 User Group is not responsible for the content of any program in its library.

Program	Programmer	Language	Type
AD&D Character Sheet	James Bainard	b/xb	AD&D(form)
AD&D Party Treasure Sheet	James Bainard	b/xb	AD&D(form)
Dungeon Master's Familiar	Douglas Howe	xb (disk)	AD&D
Mini Quest	Michael Swiridenko	b/xb	adventure
Night Ranger	James Bainard	xb	arcade
Star Fighter	Kent Degan	b/xb	arcade
Grand Prix Circuit	Michael Swiridenko	xb	arcade
Helio Copter	Paul Degner	xb	arcade
Zap	Sheldon Itscovitch	assembly	arcade
Apollo Lander	Michael Swiridenko	xb	arcade
Lunar Lander	Kent Degan	b/xb	arcade
Checkers	Michael Swiridenko	b/xb	board
Checkers	Kent Degan	b/xb	board
Chess	Michael Swiridenko	b/xb	board
Blackjack	James Bainard	xb	card
Blackjack	Kent Degan	b/xb	card
Poker	Kent Degan	b/xb	card
Whist (2 handed)	James Bainard	xb	card
Craps	Kent Degan	b/xb	casino
Craps	Michael Swiridenko	b/xb	casino
Multi File	James Bainard	xb	database
Tic Tac Toe	Michael Swiridenko	b/xb	game
3D Tic Tac Toe	Kent Degan	b/xb	game
Eliza	Kent Degan	b/xb	game
Maze Generator	Kent Degan	b/xb	game
Quest	James Bainard	xb	game
TI Mastermind	James Bainard	b/xb	game
Battle of the Sea Lords	James Bainard	xb	modemgame
Bounce	Kent Degan	b/xb	physics
Dipole Antenna	Kent Degan	b/xb	physics
Log Dipole Antenna	Kent Degan	b/xb	physics
Loop Antenna	Kent Degan	b/xb	physics
Yagi Antenna	Kent Degan	b/xb	physics
Yagi Antenna (no. 2)	Kent Degan	b/xb	physics
Decoder	James Bainard	xb	puzzle
Hangman	James Bainard	b/xb	puzzle
Tower of Hanoi	James Bainard	b	puzzle
Loan Amortization	Kent Degan	b/xb	utility
Lowercase Letters	James Bainard	b/xb	utility
Month Calender Generator	Michael Swiridenko	b/xb	utility
Phone Directory	Michael Swiridenko	xb (disk)	utility