

065
8811



The Ottawa T.I.99/4A Users' Group



VOLUME 7 NUMBER 09 NOVEMBER 1988

ALIVE AND WELL!

RE MEM BER?

FRIDAY
OCT 28
1983

GRAPHX FUNNELWEB TELCO
SUPERCARTS PC-TTRANSFER
TI BASE GRAMKRACKER
FONTWRITER MACRO ASS
SPAD XIII RAMDISKS FORTH
PRINTER'S APPR. ARCHIVER
GENEVE PREDITOR DM1000
TI-ARTIST FASTTERM
ADV. DIAGS JOYPAINT
PBASE LEGENDS
DISK UTIL. BATCH-IT
C99 HDCC 80COLCRD
EXPLORER ... ETC.

**Come to the meetings to learn all about
new software, and get
some hardware at our FALL AUCTION**

DON'T FORGET THE MEETING -- November 1, 1988
It's auction time! Bring your goods, your money, and your friends!

P.O. BOX 2144, STATION D, OTTAWA
*** ONTARIO, CANADA K1P 5W3 ***

COMING EVENTS

November Meeting: November 1, 1988 Merivale High School
7:30 p.m. Bring your goods and
money! It's auction
time!

Graphics Workshop: November 19, 1988 Merivale High School
(Featuring GRAPHX, 10:00 a.m. Contact Bill Sponchia for
TI-Artist, and further information.
Joypaint)

December Meeting: December 6, 1988 Merivale High School

4th Annual TI-FEST April 29, 1989 Merivale High School
Contact Jane Laflamme
for details

Newsletter Deadline: November 15, 1988 (or any time before that!)

EDITOR'S NOTES

from
Ruth O'Neill

This fall is shaping up to be an exciting one, with new hardware and software on all fronts. Myarc's Hard & Floppy Disk Controller has been shipping for a while, now, and it looks to be one of the most significant hardware items to be released to date. In software, there are some exciting products just recently released or coming soon. The anxiously-awaited FirstBase by Warren Agee is at last available from Genial Computerware. Look for a review of this new database manager in coming issues! Genial has also released a program by Wayne Stith called "TRIAD" - a disk manager, terminal program, and 40-column editor in one. An assembly-language tutorial, similar to the celebrated KWIKFONT tutorial, is planned. (Wayne recently contacted me to tell me that TRIAD now works on the Geneve as well as the TI.) Genial TRAVELER 2 & 3 of volume 2 are out, too.

Asgard has released two exciting products from our own members, with one still to come. PrEditor, an excellent programmer's editor, was written by Tom Bentley and released at the recent Seattle fair. Batch-It, a collaboration between Tom Bentley and Charles Earl, was released there as well. Batch-It is a batch programming language for the TI and Geneve that has to be seen (and used) to be appreciated! Look for both these fine packages at the meeting.

I've now seen two issues of ASGARD NEWS, a new publication, and I must say that I'm impressed. Its focus is on "news, views, & reviews", rather than programming or building hardware. As such, it is an exceedingly worthwhile endeavour, and I wish Chris Bobbitt, the editor, all the best with it. It is currently a quarterly publication, and the last subscription information I have is a special of \$9.00 for 4 issues (+ \$3.00 for Canadians). To subscribe, send your money and address to:

ASGARD PUBLISHING
Box 10697
Rockville, MD 20850

I'll have to admit that I could be a bit biased... the second issue did contain an article by Ruth O'Neill, but I enjoyed the rest of it every bit as much as seeing my own work there. Asgard News pays for articles, by the way, so if you have ideas along that line, contact them for that, too.

In closing, I'd like to welcome the new executive, especially the new members, for getting involved, and those continuing in new positions, for sticking with us. Please give them all the support they deserve -- and that's a lot! Get involved! It's the best way to truly enjoy your club!

The President's Two Cents' Worth
by Jane Laflamme

First and foremost, thank you Peter Arpin and John O'Connor for giving two years of your life to our club; your input has been and WILL be invaluable. Welcome and congratulations to the new executive members, Al Palmer (Vice President) and Marcelle Gibson (Secretary). Welcome back Bill Sponchia (Treasurer), and Michael Taylor (Past President). To our committee heads, thank you for your hard work in the past year -- we look forward to working with you in the coming year. To the general membership, if you have any time to assist, even an hour here and there, we will welcome you with open arms. (You will be surprised how much you will receive back in knowledge and friendship!) Executive meetings will be held on the second Tuesday of the month as a rule, and anyone is welcome to attend, but please let us know beforehand.

Regrettably, we felt we were unable to hold the membership fees at the same rate for another year. Membership fees for 1989 will be increased to \$25. (If I had been smart, I would have postponed this announcement at least a month. This is NOT a good way to start a Presidency!).

There will be a delegation of six Ottawa members attending the Chicago Fair on November 12th this year. If you would like to attend, I believe we may still be able to make arrangements for you, so please give me a call right away.

And finally, I was going to change this column title to "Presidents Sixth Cents" (you know, women and their intuition?..!) but the person who coined the title threatened to cry if it was changed, and I hate to see a grown man cry....

Until next month...

Club News and Business

We will continue to hold the raffle at the monthly meetings. The next prize will be a new piece of software, hopefully (if it is received in time) Batch-It, written by our very own Tom Bentley and Charles Earl and released by Asgard Software. What a marvel this item is!

Bill Sponchia is starting up another beginners' Assembly Language class, (I think that means the current Beginners will now be promoted to Intermediate! I congratulate you on your promotion.) Please see his column elsewhere in the newsletter, and contact him if you are interested.

On our next monthly meeting on Tuesday, November 1st, we will be holding an auction. Bring your unused treasures; someone will sure to have been looking for one of them. Also bring CASH. As we will be inviting the general public to our meeting, we are asking for cash sales only (unless the seller is your best friend and will accept your cheque). We will also accept reserve bids. The final sale will be a contract between the buyer and seller; the club will give you the opportunity to buy and sell, and will not be taking a percentage of the sales. This type of activity has been very popular in the past, so talk it up to those members that haven't come out to see us lately, and invite your friends.

We're also expecting to see Bob Boone at the November meeting, so if you've been looking to buy some new hardware or software, or just a chance to see him, come on out! Bob hasn't missed a meeting, by the way, since he was transferred to North Bay.

We hope to have the Newsletters we have received from other groups at our next meeting. I say "hope" because they will have to be numbered and catalogued, and a system of control will be made up. If this gets done by the next meeting, they will be there for you to sign out and read at your leisure. These newsletters are a great resource for our group, so please treat them with care and respect.

THE OTTAWA TI-FEST CONTINUES!

The date has been set for our Fourth Annual TI-FEST, and the rooms have been tentatively booked. (All this means is that they don't consider it booked until we give them a down payment.) It will be held at Merivale High School in the Cafeteria, two classrooms, and the lecture hall, on Saturday, April 29th. This is a very attractive date as it is getting warmer at around that time, and the hotel accommodation is still at the non-tourist-time rate, i.e. LOWER! The official hotel is the Talisman once again. For those of you who receive this Newsletter in different parts of Canada and the United States, think of getting a delegation to come to our fair. Ottawa have attended, en masse, several fairs including Chicago and Boston, and have had a great time. CIM-99, Montreal, even hire a bus to come and see us. Think about it. We hope once again to draw high profile TIers and supporters as we have in previous years.... It is one great time, meeting people, sharing your hobby, etc. Watch the Newsletter for further developments.

Saturday Seminar by Bill Sponchia

By the time you read this the October Saturday Seminar (October 15th) on PRBASE will have been completed. If it went as well as September's on Multiplan it was a success.

November's Saturday Seminar will take place on November 19th and will have as its theme "Graphic Programs". Lucie Dorais, of Newsletter cover-page fame,, has graciously accepted my invitation to conduct the session.

She has informed me that participants of this Seminar will learn (and I quote) "How to use GRAPHX, TI-ARTIST and JOYPAINT and some of their not often thought of uses." Another of the interesting areas touched will be "How the best features of each can be used to work on one image..."

Let's see all you budding artists (or those who have no talent but want to try anyway).

Date: Saturday, November 19th
Time: 10:00 AM
Place: Merivale High School
(room number as yet unknown)

Presently I have only one more topic to be presented as a Saturday Seminar. The month it will be held has not been fixed as I am waiting for a "best time" from the co-ordinators -- Charles Earl and Tom Bentley. Needless to say, both are quite busy -- Tom with his new addition to the family (by the way Tom, a belated congratulations; to your wife also because I have a feeling that you can't take full credit for this -- she probably helped out); and to both Charles and Tom for their new program releases through Asgard, namely Batch-It, PrEditor and Press (congratulations guys and also a big thanks).

The Seminar will be on the various utilities and tools presently available to us.

As I stated above this is the last topic on my list and I don't have any other firm ideas. How about helping out -- let me know what you would like to see presented. It doesn't have to be a topic that would take the full day (we have the room from 9 to 3) because if we get two or three small items we will just fit them into one day.

Anyone interested in having an Extended Basic (or just Basic) workshop? How about a "Programming" workshop? HELP! I need ideas.

Members outside the Ottawa area (Montreal, North Bay, Sudbury, Peterborough, etc) would you like to attend? You're more than welcome. Just let me know what you would be interested in.

BROWSING THE LIBRARY
--with DAVE MORRISON

To open, let me thank Lucie Dorais for her assistance in preparing the October Disk of the Month. To be completely honest, all I did was make the label and make a number of copies! I also wish to thank Art Green, Stephen Bridgett and Lucie for supplying the Library with new programmes in October.

Considering the relatively small turn-out for the October meeting, sales of disks were quite brisk, even though we had a demonstration (peaceful!), an auction (one item!) and an election (uncontested!).

I have not yet decided on the contents of the November Disk of the Month, but I'll have the full stock of all disks that have been offered during the past several months plus Art Green's MACRO ASSEMBLER (three SSSD disks), Clint Pulley's C99REL4A, and TI-TOOLS-2. The programmes by Art Green and Clint Pulley are beyond my comprehension and I have not yet performed the necessary de-archiving of TI-TOOLS-2, although even I could assume that it is probably an up-date of TI-TOOLS-1! Incidentally, all of these programmes will be available in your choice of single or double sided disks.

Last month, I said that the Library column had to improve, but when there isn't much to write about, it is difficult to live up to my promise. I might mention that I have revised certain sections of the Library (Documents, Speech, & Graphics, and that I should be able to produce a new Library Catalogue by the end of the year. I intend to revise the catalogue to have all the Fareware material cross-referenced by subject, rather than relying on a disk-name for information. It is my intention to simplify, but in order to do so, I have to go through some complicated manoeuvres! This will, I assure you, NOT be done by the end of the year!

See you at Merivale High School on Tuesday, the first day of November at 1930hrs (7.30 p.m.!).

Beginners Assembly Workshop
by Bill Sponchia

By popular demand, another Beginners Assembly course (?) will be started up. The date of the first meeting has not yet been set but it will be sometime in late November or early December. At present, it is scheduled for a week night; however, that is subject to change if there is enough response. So far we have a commitment of two young ladies attending and one gentleman who is sitting on the fence.

Everyone is welcome, but the course will be targeted to the rank amateur (ie, we will start at square one). Here is a simple test to judge where you stand.

Give the meaning of the following:

- | | |
|------------|---------|
| a) op-code | d) BL |
| b) AI | e) VSBW |
| c) MOVB | f) REF |

If you scored 0 to 3 correct - you will be right at home
4 to 5 correct - you should assist the teacher
6 correct - you should be the teacher
7 correct - you cheated
8 or above - dummy, 1 mark per answer

Interested... give me a call.

By the way, at the end of this course (whenever that is) the participants will follow the time-honored practice of doing a group project. A firm choice has not been made as to what it will be, but the four main candidates are:

- 1 - a database program
- 2 - a spreadsheet program
- 3 - an integrated disk manager/disk editor program
- 4 - a program to put our name on the screen

A Modest Proposal Accepted
by Jane Laflamme

We have mentioned on the BBS and, I believe, in this Newsletter, that we congregate at one of our member's houses, and attend, nearly every week, a Sunday conference on CompuServe. The topics and guests are diverse. It is an enjoyable evening, as well as being extremely informative. Not only are we learning from the guest at the conference, but we also have a chance to chat with other Tiers all over the United States, and Canada, and it precipitates some very animated discussions among the Ottawa members attending the conference around the terminal.

One conference, after the topic had finished, we had some time left, and Jim Horn, the TI-Forum CompuServe SysOp and moderator, dropped an idea of his that he had been mulling around for sometime. It went somewhat this way. (Forgive me, Jim, if I am interpreting rather than quoting..)

As you know, there were literally millions of TI consoles purchased when Texas Instruments were in their heyday, and the terrible part of this story is that many have been relegated to the closet. According to T.I., there were 7,000 units sold in the Ottawa-Hull area alone, and many are sitting there doing nothing -- and probably a lot of owners believe that they are no longer supported. We as a group somehow have to reach those thousands. Some would use their machine if they knew there was support; others may have moved on to the great "Blue Machine" or another such. So what Jim proposed in short, was to seek out those unused, unwanted units, and let a teacher adopt them. This way teachers and children could have hands-on experience at a very low cost. And the units would be useful rather than just sitting there.

We discussed Jim's idea at the last executive meeting, but came to the conclusion that schools in Canada, or should I say, in the Ottawa-Carleton area are well supported as far as computers are concerned. Many schools even have computer labs and specialized computer courses. On top of that, more and more homes have computers, and therefore children do have access. The discussion turned to the opposite end of the spectrum: Senior Citizens.

Their grand-children, as well as their children, are becoming more and more computer literate. The language and buzz-words used are not in their language as a whole. Could we consider approaching an organization and see if there is an interest? There was a resounding yes, and isn't it about time we put something into our community?!

The next day, Bill Sponchia approached Sandra Presley, the Executive Director of the Kiwanian Alex Dayton Senior Activity Centre at the Carlingwood Mall. Mrs. Presley sounded very interested and asked if Bill and I would meet her the next evening to discuss our ideas. After throwing around some more ideas, we felt the following should be our course of direction at this time...

Bill would hold a class once a week, with as many console and monitor units as possible to enable the seniors to have hands-on experience. We will solicit them from our membership as loans. If the class is a success, and the interest warrants, we would then go into a full fledged campaign to request donations to the centre from both our members and the general public. At first, it would only be the consoles and monitors (old black and white TVs), but later, pieces of unused equipment you would like to donate. We could reserve some funds from the raffle we have been holding, and/or have a 50/50 draw. There are many ideas that can be pursued. We would be very interested in hearing yours.

In conclusion, we ask for loan units of consoles and monitors, be they black and white TVs, or any other. We also request volunteers to deliver and pick them up after the class. If you would like to participate in this worthwhile project, please contact the co-ordinator, Bill Sponchia, at any reasonable hour at 523-0878.

(Don't put down this newsletter and forget! Get involved!)

BETTER MOUSE TRAPPING

by Ken McKenzie

Is there really anything new out there in the software world? Or are we really trying to create a better mouse trap, or re-invent the wheel? For the sake of argument, let's agree that we are improving on the old design of the wheel, and how we are improving the wheel is by adding bolts to it so it is solidly attached. These bolts are how the wheel interfaces with the vehicle that it is attached to. This is similar to how a piece of software interfaces with a USER.

The TI programming world is still making progress with a computer that has been discontinued years ago! The mechanics of the software, the algorithms are all improving, but the way that the software deals with the user is not keeping pace. Programmers have been working like crazy producing some excellent software products, they are fast, effective, and dependable, but they aren't CUTE!

Everyone is working towards software excellence but they seem to be forgetting that other people may want to use it as well. Whether it is Commercial, Shareware, Freeware, or some other label we tack on to software that is used by more than one person, software has to be user friendly. The easier, friendlier, and prettier the interface that exists between the computer and the user, the more likely it is to be used. This can translate into real dollars in the case of commercial or semi-commercial programs.

The TI world seems to be stuck on the menuing system that is similar to that of DM1000. You are forced to page through menus selecting options that lead you to other menus that let you select options that lead you to menus that let you select... and on and on. Somewhere along the line, programmers have forgotten that key strokes are work, and if someone wants to work they will do it in the office, not at home. People don't mind pounding on the keyboard for 5000 extra key strokes if they are being paid for it.

I think that it is time for programmers to start looking at how they make software user interfaces. Wouldn't it be nice to have applications that use the Slash-key-activated menus, or the Alt-key-activated menus that are the type used by Lotus 123 and all Microsoft products? Or the ICON type menus of the Apple MACs?

Here you are using your favorite disk manager, and instead of paging through different menus you just hit the CTRL key and up pops a command bar that says FILES DISK UTILITIES. You press F and down drops a menu that asks you if you want to Catalog cOpY Delete Rename View Print a file.

This is a simple example that can be applied to any piece of software, performing any task. Whether the job is complex or simple... DataBase, SpreadSheet, or simple program, the concept of simple and effective interfaces will work. Being able to POP/JUMP/PULL will make life easier. TI-WRITER, the ancient and much used and copied actually uses a form of this type of interfacing. You hit a combination of keys and you are sent to the top command line.

Many people out there in TI land are going to say that they want to remain different and not upgrade/change the way that the TI world works. These people are the ones who want to keep everything as Texas Instruments originally designed it. To these people I say, "WHY? The machine is a computer, not an old piece of furniture that you may appreciate." If the TI is made to look and feel easier to use, if the interfacing becomes more user friendly, I think this machine can live for many more years. If it does not change, I feel that it will die a much quicker death.

I would like to hear other peoples comments and opinions -- it may be possible for the users to influence the programmers by telling them what we want. If you have any comments please write to me at:

2678 Stone Cr.
Ottawa, Ontario
K2H 6Z1

I can also be reached on TEXLINK (738-0617) as user #73.



Mr. Diskette has found a new star

Mr. Diskette is an authorized dealer and repair depot for all Star Micronics printers.

Popular printers like the very affordable Star



NX-1000 and Star NX-1000 Rainbow — the great little

printer that prints in 7 colours! Both printers handle friction or tractor fed paper as well as offering four fonts and

near letter quality type!

1 year warranty

Don't forget, you can only be assured of your full warranty and complete dealer support and service on your Star Printer if you've purchased your printer from an authorized dealer. So always check for the Star authorized dealer sticker like the one Mr. Diskette has!

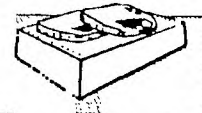


Free starter kit

Purchase a Star NX-1000 or NX-1000 Rainbow printer and receive a free starter kit. The kit includes two

printer ribbons

(value \$19.95 ea.) and a generous supply of paper (value \$7.95)!



Honest value

So now good honest value just happens to be your first reason to visit Mr. Diskette... not the only one!



Mr. Diskette locations and hours

Downtown	Nepean	Gloucester
105 O'Connor St. (just north of Laurier)	1600 Merivale Rd. (across from the Brick)	1456 Cyrville Rd. (next to Startop Rd.)
232-5203	727-0180	749-3100
Open 9 - 6, 9 - 7 Thu & Fri, Saturday 9 - 5	Open 9 - 9 Saturday 9 - 5	Open 9 - 9 Saturday 9 - 5

This Ad prepared with Mr. Diskette's desktop publishing system

**EXPANDING EXTENDED BASIC'S POWERS,
WRITING ASSEMBLY ROUTINES: PART 2**
By David Caron

In my last article, I described Extended Basic's four assembly subroutines. In this article, I will show you how to actually write an assembly routine designed to be run from Extended Basic.

When Extended Basic executes something like:

```
CALL LINK("START")
```

it will look up the name in the def table at the top of low memory, find out where the assembly routine starts in low memory, and execute it. (This much has already been explained in my last article.) When an assembly program is executed from Extended Basic, the workspace registers will be the GPL Workspace Registers. These registers are used by the GPL interpreter program. That means that any program written in GPL (Graphics Programming Language) will be using these registers. Examples of GPL programs are TI-BASIC, EXTENDED BASIC, and most of the Programs in Command Modules. (That's why the Basic Language on the TI-99/4A is so slow. Your Basic program gets interpreted by TI-Basic or Extended Basic which gets interpreted by GPL -- take a look at page 406 in the Editor/Assembler manual). Anyway, the point I am trying to make is that you might find it convenient to define your own workspace since the instant you run any assembly utilities the contents of these registers will be changed. More on this later. For now, I will talk about the assembly utilities that are built into Extended Basic.

For now I will only mention the four most important utilities, which are:

```
VSBW (Vdp Single Byte Write, starting address: >2024)  
VMBW (Vdp Multiple Byte Write, starting address: >2024)  
VSBR (Vdp Single Byte Read, starting address: >2028)  
VMBR (Vdp Multiple Byte Read, starting address: >202C)
```

As you may have guessed, these utilities refer to accessing the VDP memory. If you intend to make any worthwhile program you will find yourself using these routines quite a bit. These routines, along with others I have not yet mentioned, get loaded from the Extended Basic module into low memory when a CALL INIT is executed. CALL LOAD(disk name) will place assembly routines directly after the assembly utilities in low memory.

VSBW: To use this routine you must first indicate where in VDP memory you wish to write to. Register zero must contain this value. For instance, if you wanted to write something in the screen at position (1,1), register zero would be loaded with the value 0 since screen position (1,1) happens to be VDP address >0000. Then you must let VSBW know what ascii code you wish to write to location >0000. The most significant byte of register 1 must contain the ascii code.

(By the way, Extended basic has a funny way of displaying its characters. When you see a blank character on the screen, you probably think that it is the ascii code 32, but in reality it is the ascii code 32+96 =128. This is actually Extended Basic's fault, not the TI-99/4A's, and there is a perfectly logical explanation for this. It is slightly complex, though, so I will explain this some time in the future. For now, take it for granted that when you are dealing with ascii codes, add 96 when sending them to the VDP screen and subtract 96 when receiving them from the VDP screen.)

Now that you have indicated the address and the data, the VSBW routine can be executed with BLWP @VSBW. (I will explain this assembly instruction later for those of you who are unfamiliar with this.)

VSBR: This utility is identical to VSBW except that it peeks from the VDP memory location defined in register zero and writes the data into the most significant part of register one. If this happens to be an ascii code which was peeked from the VDP screen, you must subtract 96 from it to make any sense out of it. This routine is executed with BLWP @VSBR.

VMBW: This nice utility lets you poke more than one byte into VDP memory at once. Once again you place the VDP address into Register zero. However, a CPU memory address must be placed in register 1. When VMBW is executed, the data in the CPU address indicated by register 1 is transferred to the VDP memory address indicated by register zero. The number of bytes you want transferred is placed in register two.

VMBR: This utility is exactly the same as VMBW except that the data is now transferred from VDP memory to CPU memory.

And here it is! What you have all been waiting for: an example which will hopefully make some sense of what I have said.

```

DEF  START      *This places to word "START" in the def table
                    *along with its start address.
VSBW  EQU  >2020 *The Extended basic assembly environment has
VMBW  EQU  >2024 *no ref table so the assembler directive EQU
VSBR  EQU  >2028 *(equate) must be used to define the constants
VMBR  EQU  >2030 *VSBW, VMBW, VSBR, and VMBR, take a look at
                    *page 415-416 of the Editor/Assembler manual.
NEWREG BSS  32   *This is where our workspace will be. 32 bytes
                    *are reserved since 16 registers X 2 = 32
START LWPI NEWREG *This instructs the computer to use the memory
                    *locations indicated by NEWREG as the work-
                    *space registers.
LI    R0,>0000  *Loads register zero with the 16-bit data 0
LI    R1,83+96 *Loads register one with the 16-bit data 179
SWPB R1        *Swaps the least significant and most significant
                    *byte of register 1.
BLWP @VSBW     *Causes the computer to branch to the data at
                    *the address indicated, then return to your
                    *program.
INC   R0       *Increments the data in R0 by 1.
LI    R1,84+96 *Loads register 1 with the data 180
SWPB R1        *Swaps the bytes in R1.
BLWP @VSBW     *Executes the VSBW utility.

INC   R0       *We know what this does, now register 0 = 2
LI    R1,>A100 *Loads R1 with the indicated hexadecimal value
                    *In decimal this value is 161=65+96. Now we do
                    *not have to do a swap command.
BLWP @VSBW     *Sends the ascii code in R1 out to the screen

INC   R0
LI    R1,>B200 *Loads R1 with 82+96
BLWP @VSBW

INC   R0
LI    R1,>B400 *Loads R1 with 84+96
BLWP @VSBW

                    *These next instructions let you return to
                    *Extended Basic. See page 441 of E/A.
CLR   R0       *Same as loading R0 with the data >0000.
MOVB R0,@>837C *The most significant byte of R0 is copied
                    *to the byte located at >837C, so you end up
                    *simply clearing the byte at address >837C.
                    *This byte is the GPL status byte and must be
                    *cleared before returning to Extended Basic.
LWPI >83E0     *Go back to GPL workspace registers.
RT     *Return to Extended Basic.

```

And that's it. Can you guess what this assembly routine does? Yes! it prints the word "START" from row 1, column 1 to column 5. You could substitute START for your own name if you wished to, however I suggest that you wait until I show you how to do that using a loop. (Next article, Much less code). If you have any questions do not hesitate to call me at 745-4618.

FAST EXTENDED BASIC

LUCIE DORRIS

Horror: a bug has crept into last month's program, but Ralph Kuhn spotted it. After teaching you all about leap years for century years that divide by 400, I mistakenly programmed Tex to be right only for those years that divide by 2000... So, in line 420, please replace the two "2000"s by two "400"s. Thank you.

After two months of heavy calendaring, it is recreation time. Even if my name is not Vanna, here is my "Wheel of Fortune" (if you type it, remember that you don't need to type the comments after the "!". I add them to the text listing to make the program easier to understand):

```
100 REM LUCKY WHEEL / by L. Dorais / Ottawa U.G. / Sept. 1988
110 REM
120 DISPLAY AT(1,9)ERASE ALL:"LUCKY WHEEL"
130 OPTION BASE 1 :: DIM CH(16),R(16),C(16) :: JP=3000
140 DATA 96,5,8,104,5,10,120,6,12,137,8,14,96,10,15,112,12,15,104,
14,14,136,16,12,!(character,row,column,:::!)
150 DATA 96,17,10,137,17,8,104,16,6,112,14,4,96,12,3,104,10,3,137,8,
4,128,6,6
160 CALL COLOR(9,9,1,10,11,1,11,5,1,12,14,1,13,13,1,14,16,1)
170 GOTO 190 :: BK,BST,CHA,DEL,K,X,SC,SP,SR :: CALL HCHAR
180 CALL KEY :: CALL SOUND :: CALL SPRITE :: CALL MAGNIFY :: CALL
DELSPRITE :: CALL PATTERN !@@P-
190 AS="3C7EFFFFFFF7E3C" :: CALL CHAR(96,AS,104,AS,112,AS,120,AS,
128,AS,136,AS&"3C42AD9999A5423C007E7E7E7E7E7E00") ! small gr.
200 CALL CHAR(36,"071C306040C1839F8783C14161301C07E0380C868283C1E1F9
C18302060C38E0") ! wheel-1
110 CALL CHAR(40,"030C3020484583878783454820300C03C0300C0412A2C1E1E1
C1A212040C30C0") ! wheel-2
220 CALL CHAR(44,"000007081020202020202010080700000000E0100804040404
04040810E00000",140,""141,""142,""143,"") ! spin choice
230 FOR X=1 TO 16 :: READ CH(X),R(X),C(X) :: CALL HCHAR(R(X),C(X),
CH(X)) :: NEXT X ! display the colored wheel
240 FOR X=1 TO 5 :: DISPLAY AT(2*(X+1)+1,19):USING "# = #00":
CHR$(8*(X+1)),X :: NEXT X ! display the points
250 DISPLAY AT(15,19):CHR$(136)&" = DOUBLE" :: DISPLAY AT(17,19):
CHR$(137)&" = HALF"
260 DISPLAY AT(20,2):"GOAL:";JP;TAB(17);"BANK: 100": : : "PLEASE
ENTER YOUR GOAL AND STARTING POINTS"
280 DISPLAY AT(21,2):"BUST:";BST :: RANDOMIZE :: SP=1
290 DISPLAY AT(23,1):"::" :FPRESS <SPACE> TO SPIN"
300 CALL KEY(0,K,S) :: IF S=0 OR K<>32 THEN 300 ELSE CALL
DELSPRITE(#2) ! erase winning circle if key pressed
310 CALL MAGNIFY(4) :: CALL SPRITE(#1,36,16,68,52) :: DISPLAY
AT(24,4):"PRESS <ENTER> TO STOP" ! center wheel appears
320 CALL KEY(0,K,S) :: GOSUB 470 :: IF S=0 OR K<>13 THEN 320 ELSE
DISPLAY AT(24,4):" " ! turn the wheel until a key is pressed
330 FOR X=1 TO INT(RND*9) :: FOR DEL=1 TO X*20 :: NEXT DEL ::
GOSUB 470 :: NEXT X ! slow down and stop
340 CALL HCHAR(R(SP),C(SP),CH(SP)) :: CALL DELSPRITE(#1) ::
SR=8*(R(SP)-1)-3 :: SC=8*(C(SP)-1)-3 :: CHA=CH(SP)
350 CALL MAGNIFY(3) :: CALL SPRITE(#2,44,16,SR,SC) ! winning circle
360 FOR X=1 TO 15 :: CALL PATTERN(#2,140,#2,44) :: NEXT X ! flash
370 IF CHA=137 THEN BK=INT(BK/2) :: CALL SOUND(200,-3,0) ::
GOTO 420 ! half the points
380 IF CHA=136 THEN BK=BK*2 ! double the points
390 IF CHA<136 THEN BK=BK+100*(CHA/8-11) ! points 100 to 500
400 CALL SOUND(100,784,0) :: CALL SOUND(100,1047,0) :: IF CHA<136
THEN 420 ! bells for all wins
410 CALL SOUND(100,880,0) :: CALL SOUND(150,1568,0) ! more if double
420 DISPLAY AT(20,22):BK :: IF BK>BST AND BK<JP THEN 290 ! continue
430 IF BK>=JP THEN AS=" YOU REACHED YOUR GOAL!" ELSE AS="SORRY! YOU
LOST EVERYTHING!" ! end game
440 DISPLAY AT(22,1)BEEP:AS: : " ANOTHER GAME? Y:N"
450 CALL KEY(0,K,S) :: IF S=0 OR(K<>78 AND K<>89) THEN 450 ELSE
IF K=89 THEN CALL DELSPRITE(#2) :: GOTO 260
```

```

460 END
470 CALL PATTERN(#1,40) :: CALL HCHAR(R(SP),C(SP),CH(SP)) :: SP=SP+1
   :: IF SP=17 THEN SP=1
480 CALL PATTERN(#1,36) :: CALL HCHAR(R(SP),C(SP),138) :: RETURN

```

The main point, when programming a game, is to plan it well: decide what you want, and find the easiest or most elegant way to make the program run as fast as possible. The first decision here was to determine the number of "spikes" in the wheel: each value obviously needs its own color, but we want to be able to display some text, so character sets 1 to 8 are no-no's. We are left with sets 9 to 14 (sets 15 and 16 cannot be used in XB). That is six colors, not enough for an exciting game... but the lowest values can be repeated, and one color can be used for two values: double and half the amount in the bank.

Using a screen editor, I played with the little spots until I found that 16 spots made a perfect circle. The next step was to decide how to place the spikes according to their point value, since the lower values must appear more often than the higher ones, and the values most often found on the wheel must be evenly spaced.

To respect my self-imposed limit of one page, and because XB limits our use of colors and graphics, I greatly simplified the design of the lucky wheel: instead of a real turning wheel, mine is fixed; to simulate the actual spinning, a white square quickly moves over the colored spots that represent the spikes. A couple of sprites create a real turning wheel in the center, to make the illusion look a bit better. And when you press the button to "stop the wheel", the little square slows down until it stops for good. Your winning spot is then enhanced by a flashing circle, and you win or lose, with appropriate bells and whistles.

To simplify the calculations, the character values are kept in an array, as well as their row and column positions; this is the DATA in lines 140-150. To modify the values of the spikes, just change their character values; they range from 96 (100 points) to 136 (double points), with a step of eight, since we can use only one character per set: 96, 104, 112, 120, 128, 136.

Initialization, including the pre-scan, is done by lines 160-220; the JP variable in line 130 is for the default "jackpot", i.e. the goal you want to reach; you may change that value at the beginning of each new game. The six colored spots are defined by A\$; character 136 also defines 137 as an empty spot with a cross (half your points), and 138 as the revolving square. Lines 200-210 are for the two sprites of the center wheel; line 220 defines a circular sprite, and we empty characters 140 to 143 to get an empty sprite that will alternate with the circle to flash around the winning spots.

We next display the game screen, the lucky wheel in line 230, and the point values in lines 240-250. Note how I solved the problem of displaying five similar lines for the values 100 to 500, much shorter code than five "DISPLAY ATs". The program then asks you to enter your final goal (JP) and the starting amount of points in your hand (BK). Your "busting" points BST will be half your starting amount. Finally, the variable SP, for spin, is reset at the beginning of each new game, as is the randomizing factor.

The variable SP is the most important one of the game: Tex uses it to spin the little square around the wheel (see the sub in line 470: SP is increased by one, and when it reaches 17, it is reset to one because we have only 16 spots; note also how the GOSUB statement in line 320 is placed inside the CALL KEY one, for fast action). As soon as you "spin", a turning wheel appears in the center of the lucky one; the CALL PATTERNS in the sub make it appear to turn for real. With a magnifying factor of four, our sprites, made up of four characters, are displayed by 16, thus making them double in size.

In line 330, Tex adds a little chance factor: since stopping the game right when you press <ENTER> would make it too easy for you to cheat, I introduced a random factor: after you stop the wheel, the square will travel over a few more spots, this being determined by the RND function; INT(RND*119) will be any value between 0 and 8, so you can, by chance, stop where you are, or travel as far as half the wheel. The FOR-NEXT is a DELAY played at each step, and it gets longer and longer as the value of X increases, adding to the simulation.

As long as the square travels around the wheel, the value of SP is incremented; when it stops, the value of SP, between 1 and 16, gives us the position of our winning spot in the wheel. By reading the arrays for that value, we get its character, row and column values; we can then calculate the sprite-row SR and sprite-column SC that we need for our flashing circle (three pixels up, three left, of the winning spot). But we must now get rid of the center wheel: since we want to show the winning circle "au naturel", it needs a magnifying factor of three, and our wheel's size and screen location would suffer greatly. For the same reason, in line 300 above, we delete the circle sprite #2 each time we display the bigger sprites of the wheel.

We finally win or lose points according to the CHA value of our stopping spot. If we land on a "crossed spot", our bank is cut in half, with a honk (the integer avoids fancy decimals). On the other hand, our points double if we hit the only white spike of the wheel. Line 390 is, I think, elegant coding: instead of five IFs-THEN, we derive the points 100 to 500 from the value of the character (remember, they are from 96 to 128, with intervals of eight). I'll let you analyze the mathematical formula.

The game ends if you reach your goal (easy) or if you are busted (nearly impossible). But you can always start another game!

TI BASIC continued from October
by Steven Shaw

CHAPTER EIGHT

MODULES

TI produce a number of powerful and useful modules in addition to the range of games you may be familiar with. This section is intended to help you obtain good value from the modules.

Some of the games modules contain a TEST MODE, which was inserted by the programmer to permit program debugging. The programs concerned are in machine code. Titles spotted so far are: TI INVADERS, MUNCH MAN, ALPINER and CHISHOLM TRAIL.

This is not an advertised function and may be removed. But it is worth trying with any game module you have.

Insert the module and select the game.

When the first game title page appears, quickly hold down SHIFT and press 8, 3, and 8.

A new screen should appear with various prompts. Enter responses as quickly as you can. The prompts allow you to enter the game at any level, and in the case of TI INVADERS for instance, to select a slow speed.

PERSONAL RECORD KEEPING

The Personal Record Keeping module enables you to store up to about 10k of data, and provides various handling and output facilities to help you manipulate your records.

A printer is useful but not essential. If you have a printer, greater flexibility of display, as well as additional functions, are provided with the Personal Report Generator module, which requires data prepared with the Personal Record Keeping module.

The PRK sorting routines are slow. Data is saved in memory image ('program') format, and thus uses less tape (or disk) space, is faster to save and load, and the verify option is available for tape files.

You will not be able to catalogue a collection of six thousand records, but small collections can be catalogued with the module. The number of items depends on how many characters you wish to use to describe each item.

It is possible to use the PRK module as a simple diary system, or as a very simple spreadsheet, as it is possible to perform mathematical operations on the data you place in your module.

The Personal Record Keeping and the Statistics modules both extend the range of commands available in TI BASIC. Again this is not advertised, and may be amended.

With either module inserted, select TI BASIC. You may now use the following commands:

```
DISPLAY AT  
CALL D(R,C,L,V)
```

Where R and C are the row and column the word is to start at.
L is the length of screen to be blanked from position R,C and also sets the maximum length of the display
V is a value or string or variable to be displayed.

If V is longer than L, the display will be curtailed.
R,C and L may be numbers or numeric variables.

Try:

```
CALL D(10,4,5,1/3)
```

```
ACCEPT AT  
CALL A(R,C,L,F,A,MN,MX)
```

R,C and L are as with CALL D, but in CALL A, L sets the maximum length of the input.

F MUST be a NUMERIC VARIABLE. It takes a value of 1 if ENTER is pressed, and other values if some control keys are used eg BEGIN:6 REDO:4 AID:3 BACK:7 CLEAR:2

A is the numeric or string VARIABLE to be filled with the input.

MN and MX are optional when using a numeric variable, and set the minimum and maximum acceptable values: any input outside these values is rejected.

NB: The CLEAR key is used to clear the input field. It WILL NOT break into the program! Use CALL A with a little care if you think you may need to BREAK the program!

Other commands are also added to TI Basic with these two modules, eg CALL P (partitions memory), CALL L and CALL S which save and load data in program format to the partitioned area, and CALL G which handles the data in the partitioned section and CALL H which defines the format of the data.

A booklet on these commands has been published by TI, and you may be able to purchase a copy from the main UK User group. Sample programs may be found in 'TIDINGS' Vol 2, No 4, from the main user group.

These extra commands are the only way in which a user can save data in PROGRAM format - used by most TI Modules. Program format permits tape verification, and uses a lot less space on your tape or disk.

TI-WRITER

TI-WRITER is TI's word processing module. It comes in the form of a module, a disk, and a large manual. The 32k memory expansion is required in addition to the disk drive and a disk controller, plus the Expansion Box, RS232 card and a printer.

TI-Writer is a very powerful word processor and can carry out most of the tasks a purpose built word processor is capable of.

A word processor is a great deal more than an electric typewriter. Numerous editing facilities are provided to enable the text to be manipulated.

Typical facilities (found on TI-Writer) are:

Full screen editing-

The cursor can be moved in a number of ways, using preset tabs, word tab, block movement (24 lines at a time) and the usual cursor control keys.

Text can be deleted or inserted.

Movement of paragraphs to different places in the text.

Ability to merge files, and save parts of files.

Ability to change one word in the text to another, wherever it may occur (for instance, replacing 99/4a with 99/4A).

Mailing list option: now you can write personalised circular letters (the sort which begins: Dear Mr Smith, You have been selected from the people in Acacia Avenue.....)

HOTLINE NUMBERS

The executive has expressed a desire to assist all members should you have some problems or questions, want to do some library swapping or borrow a book. This will be the place to look. Listed here are the members of the executive, committee heads, and others in the group willing to help in their specialized areas. Of course, if you wish to be placed on the list, just give me a call. I know there is a lot of expertise within our Group, so I hope to add to this list. Please respect normal hours unless you specifically know that someone doesn't mind a call at 3am, or use the BBS to leave a message at 738-0617, 24 hours a day, 7 days a week.

JANE LAFLAMME.....	PRESIDENT.....	(H) 837-1719 or (W) 745-2225
AL PALMER.....	VICE PRESIDENT.....	594-9216
MARCELLE GIBSON.....	SECRETARY.....	233-2384
BILL SPONCHIA.....	TREASURER.....	523-0878
MICHAEL TAYLOR.....	PAST PRESIDENT.....	831-0143
PETER ARPIN.....	SYSOP.....	523-0017
RUTH O'NEILL.....	NEWSLETTER EDITOR.....	234-8050
TONY HOPKINS.....	ADVERTISING.....	746-4463
DAVE MORRISON.....	LIBRARY CHAIRMAN.....	737-4889
JACK McALLISTER.....	CASSETTE LIBRARY.....	225-6989
HENRI MONAT.....	ARCHIVES.....	824-0941
LUCIE DORAIS.....	MEMBERSHIPS.....	232-0393
BOB BOONE.....	HARDWARE/SOFTWARE.....	(705) 476-0265*
ART GREEN.....	ASSEMBLY HELP.....	837-1955
DICK PICHE.....	TECH.....	521-8667
CLUB BBS.....	SET MODEM TO 8N1.....	738-0617

Bob can be reached through his sister in North Bay at the number above. If you like, you can also reach him on Delphi (CDU) or on Compuserve (73657,3617). Watch for a more permanent number here.

1989 RENEWAL \$ 25.00	NEW MEMBERS \$ 25.00
--	---------------------------------------

NAME _____

ADDRESS _____

CITY _____ PROVINCE/STATE _____

POSTAL CODE _____ TELEPHONE (____) _____

Please make cheque payable to the Ottawa TI-99/4A Users' Group and send it, along with this form, to the address shown on the cover page -- or better still, bring both to a meeting.

DEALER ENQUIRIES WELCOME

DATA*PORT
2846 Göttingen St.
Halifax, N.S.
B3K 3E1
(902) 454-0232
Data (TEXLINK) 455-2076

COMPUTER DOWNLOAD UNLIMITED
126 Sage Rd.
North Bay, Ontario
P1A 1A4
(705) 476-0265
TEXLINK BBS T.B.A.

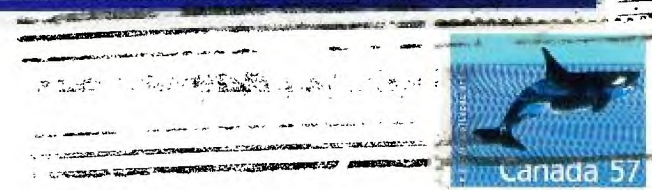
CANARIA DATA INC.
264 Weber St. W.
Kitchener, Ontario
N2H 4A6
(519) 578-3873

Authorized Canadian Distributor for Myarc, Inc.
Support your Canadian dealers:

HARD DISK CONTROLLER CARDS ARE IN STOCK NOW!

LAFLAMME & WRIGLEY DATA (Phones): Ottawa BBS (TEXLINK) (613) 738-0617
DELPHI JANLAFLAMME (No space)
CompuServe "76046,2006"

LAFLAMME & WRIGLEY WHOLESALÉ
5480 Canotek Road, Unit #16
GLOUCESTER, ONTARIO K1J 9H6
(613) 745-2225



FROM

P.O. BOX 2144, STATION D, OTTAWA
*** ONTARIO, CANADA K1P 5W3 ***

EDMONTON 99er USER'S GROUP
P.O. BOX 11983
EDMONTON, ALBERTA
T5J 3L1