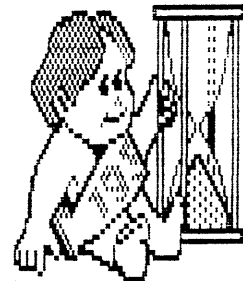
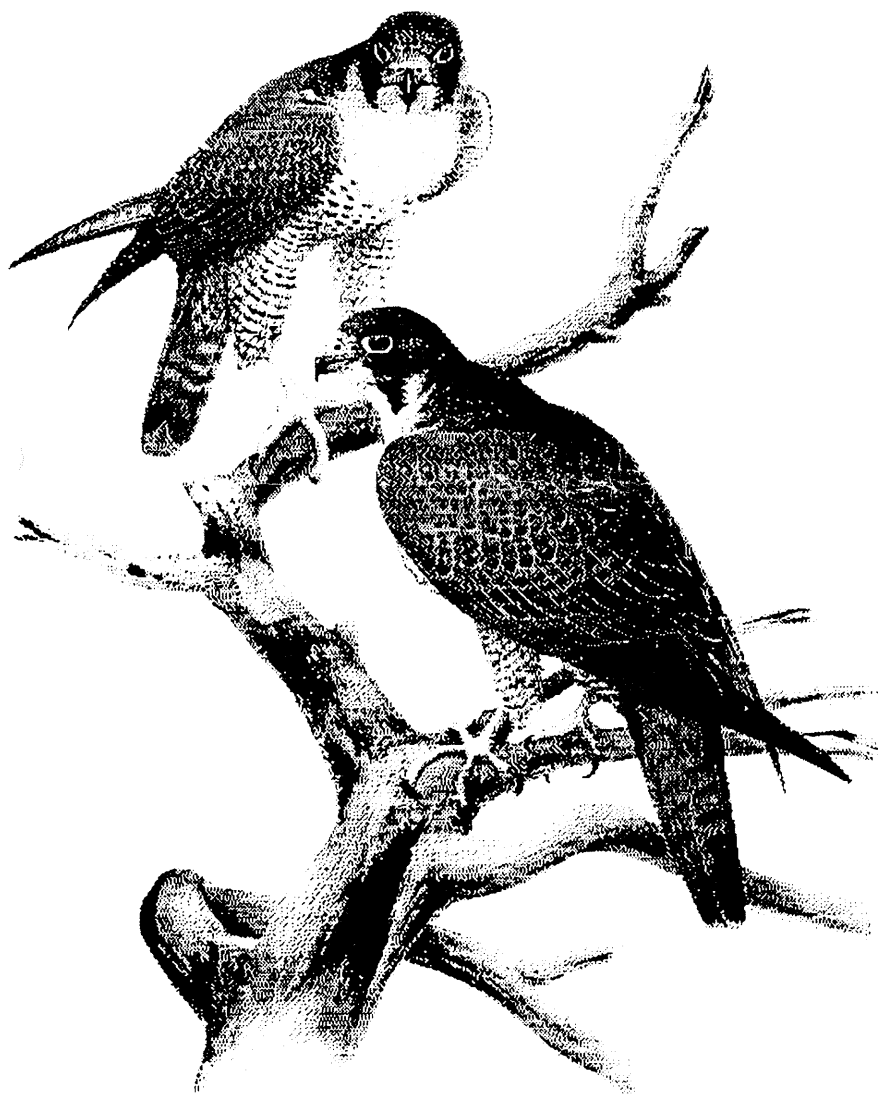


# SINC - LINK



## JAN-FEB '92 VOL 10 NO 1

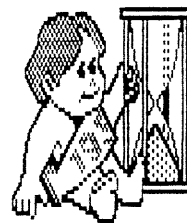


### 10th Year Issue

page 2	Club Stuff
page 3	Editorial
page 4,5	Bob's Notebook (2068)
page 6-10	Printer I/f's & Driver Software (2068)
page 11	Gold Card Update (QL)
page 12,13	Steve Gunhouse Writes (2068)
page 14	QL Tid-Bits (QL)
page 15	Mystery Program & Scroll Clear (ZX81/TS1000)
page 16	Open Letter on Disks
page 17	Timachine Fix (2068)
page 18	Larken NMI-F Key (2068)
page 19	QLips (QL)
page 20,21	ZX91 Newsletter (ZX81/TS1000)
page 22	Toolkit II Goodies (QL)
page 23-26	Steve Writes More (2068)
page 27	TorQLib D-Dump (QL)
page 28	BYTE POWER Advertisement

## TORONTO TIMEX-SINCLAIR USERS CLUB

# SINC - LINK



## JAN-FEB '92 VOL 10 NO 1

SINC-LINK IS A PUBLICATION OF THE TORONTO TIMEX-SINCLAIR USERS CLUB AND IS ISSUED 6 TIMES A YEAR. COPIES OF THE NEWSLETTER ARE \$1.50 EACH FOR NON-MEMBERS. CLUB MEMBERS RECEIVE FREE COPIES AS PART OF THE \$20.00 ANNUAL MEMBERSHIP FEE. A NEWSLETTER SUBSCRIPTION ONLY IS AVAILABLE FOR \$12.00.

NEWSLETTERS ARE EXCHANGED, FREE OF CHARGE, WITH OTHER TIMEX-SINCLAIR USERS GROUPS.

PLEASE CREDIT THIS PUBLICATION AND THE AUTHOR IF YOU COPY MATERIAL.

THE CLUB MEETS ON THE FIRST WEDNESDAY OF EACH MONTH AT FOREST HILLS COLLEGIATE INSTITUTE, 730 EGLINTON AVE. W., TORONTO.

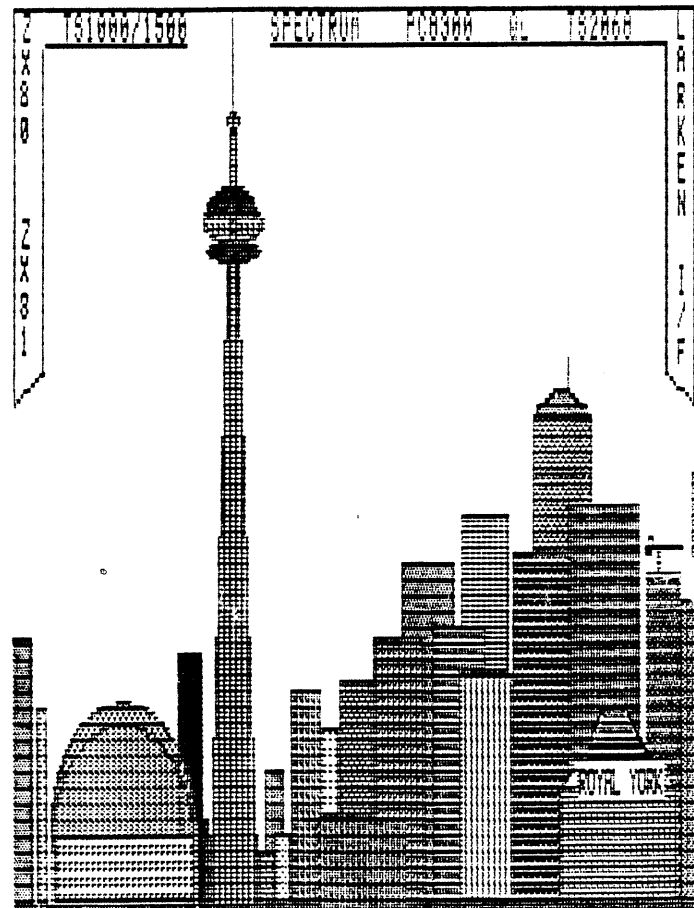
SINC-LINK IS PRODUCED ENTIRELY ON SINCLAIR AND TIMEX-SINCLAIR COMPUTERS.

SEND CORRESPONDANCE TO:

Attention: SINC-LINK EDITOR  
TORONTO TIMEX-SINCLAIR USERS  
CLUB, 14 RICHOME COURT,  
SCARBOROUGH, ONTARIO,  
CANADA M1K 2Y1

EXECUTIVE OFFICERS:

PRESIDENT:	RENE BRUNEAU ( 531-9749 )
TREASURER:	BILL LAWSON ( 444-8772 )
SECRETARY:	GEORGE CHAMBERS ( 751-7559 )
ACTIVITIES:	LOUIS LAFERRIERE ( 820-3725 )
QL CONTACT:	HUGH HOWIE ( 634-4929 )
NEWSLETTER:	JEFF TAYLOR ( 244-8583 )
LIAISON OFFICER:	GEORGE CHAMBERS, 14 RICHOME COURT,
( Out-of-town members )	SCARBOROUGH, ONTARIO, M1K 2Y1
	( 416-751-7559 )



TORONTO TIMEX-SINCLAIR  
USERS CLUB

## TORONTO TIMEX-SINCLAIR USERS CLUB

## Editorial

Welcome to the 10th anniversary issue of Sinc-Link. It's hard to believe that it will be ten years since the club first met to compare notes about the ZX81. A lot of Timex-Sinclair magazines, newsletters and clubs have come and gone but the Toronto group is still going strong thanks to its loyal local and out-of-town members and contributors. I think everyone associated with the club should give themselves a pat on the back. Well done TSers! Special thanks to George Chambers who has had articles in Sinc-Link right from that first issue up to the present and has been a member of the executive in one capacity or another since the club was formed.

## Newsletters

The past year has actually seen the emergence of some new publications like *ZXir QLive Alive!*, *FDD Newsletter*, *International QL Report* and I'm happy to report, even a new newsletter for the ZX81/TS1000 called *ZX91!* (See a sample in this issue). Next issue I will print the names and addresses of the groups with which we exchange newsletters. It's important to see at least a few of the other newsletters, particularly for the out-of-town members who don't get to see our exchange copies. There's still lots of good reading out there and none of the subscription rates are expensive.

## Swap Meet

The club meeting on Wednesday, February 5th, will be a swap meet of computer-related stuff. Do you have anything you'd like to sell/trade? Are you looking for that special piece of software? Well come on out and maybe you'll find/sell it. Out-of-town members are invited to send lists of what they would like to move or what they are looking for. A small word of caution, this is not a computerfest, so don't plan on flying across the continent to find great TS deals. This is simply a small local get-together, but if you are within, say, the golden horseshoe area, we'd love to see you. Plan on attending.

That's all for now...

J.T.

## General

Some time ago, I found out about a book called "THE SECRET GUIDE TO COMPUTERS".

The fourteenth edition 1990 by Russ Walter is available from the author himself, H.Howie, or Peter Hale of EMSOFT P.O. Box 8763, Boston MA. 02114-8763.

The Secret Guide consists of over 600 pages of general information on computers and is well worth the price of \$15.00 U.S. Dollars. For more information contact Hugh Howie or Louis Laferriere.

Maybe even our library should obtain a copy.

Louis Laferriere

2305 South Millway Apt. 409  
Mississauga, Ontario,  
Canada, L5L 3P8

# BOB'S NOTEBOOK

Just before I got my new 24-pin printer with NLQ and a host of other good features, I had written this routine to make my Fastext 80 pretend it was fitted to do NLQ printing. It does a fairly good job and I am sure it can be improved upon. Now that I have graduated to the new printer, its attraction for me has diminished somewhat but for others it may fill the bill: Type this in, save it and then run it. There are a few added notes at the end but the REM lines serve as simple instructions.

## LISTING

```

10 REM xxxxxxxxxxxxxxxxx (16 x's)
12 RANDOMIZE USR 100: OPEN # 3,"lp"
15 LET prog=PEEK 23635+256*PEEK 23636: LET prog=prog+5
20 RESTORE 50
30 FOR i=1 TO 16: READ v: POKE prog+i-1,v: NEXT i
50 DATA 33,0,130,1,0,126,62,32,119,35,11,120,177,32,247,201
60 REM *****
100 REM Tasword Special Loader to simulate NLQ Copy in Pica.
110 REM Use Single Sheet Paper.
120 REM Line 170 will switch in UK font if REM removed.
124 REM If the last line on a page does not double print
press the On-line switch once.
126 REM When continuous line feeds occur, use BREAK to STOP
action.
130 REM *****
140 RANDOMIZE USR prog
150 RANDOMIZE USR 100: POKE 16090,69
160 GO SUB 270
170 REM GO SUB 250: LPRINT CHR$ 27;CHR$ 82;CHR$ 2: GO SUB 260
175 POKE 23658,0: CLS
180 GO SUB 280: RANDOMIZE USR 100: LOAD n$+".CT"CODE
185 DIM a$(65): LET j=1
187 INPUT "start line # ? ";sl: LET sl=sl-1: LET a=sl*64+33280
188 PRINT #0;"BREAK=STOP"
190 ON ERR GO TO 400: FOR i=a TO 53000
200 IF PEEK i<32 OR PEEK i>122 THEN LET a$(j)=CHR$ 32: LET j
=j+1: GO TO 230
210 LET a$(j)=CHR$ PEEK i: LET j=j+1
220 IF j>64 THEN RANDOMIZE USR 100: POKE 16092,0: LPRINT a$:
RANDOMIZE USR 100: POKE 16092,10: LPRINT a$: LET j=1: DIM a$(65)
230 NEXT i
240 STOP
250 RANDOMIZE USR 100: POKE 16093,32: RETURN
260 RANDOMIZE USR 100: POKE 16093,0: GO SUB 270: RETURN
270 RANDOMIZE USR 100: POKE 16094,10: LPRINT : RETURN
280 INPUT "Tasword file name (<=6) "; LINE n$: GO SUB 290:
RETURN
290 INPUT "drv? ";drv: RANDOMIZE USR 100: GO TO drv: RETURN
400 ON ERR RESET: INPUT "1=more 0=quit ";mq: IF mq THEN
RANDOMIZE USR prog: GO TO 180
410 GO SUB 290: RANDOMIZE USR 100: NEW
490 STOP
500 GO SUB 290 : RANDOMIZE USR 100: SAVE "tasnlq.Bb" LINE 10

```

```
510 STOP
999 REM test line follows:
1000 FOR i=33280 TO 34000: PRINT i,PEEK i;TAB 22;CHR$ PEEK i:
NEXT i
```

Line 10 will house the code that makes this work. It will get its data from line 50 and note line 15 which finds the start of the program area to make sure the machine code in line 10 will really work.

Line 170 may not be of any value to you but if you want it to work, first check your printer manual and ensure that the code for the UK font is correct; 27, 82, 2 had to be changed to 27, 82, 3 for my 24-pin printer.

The main routine starts at line 140 which allows you to load a tasword file and choose a start line in that file. Then all the slight of hand to simulate NLQ takes place from line 190 to 230.

There are some sub-routines at lines 250 to 290, which handle the switch to UK font and back without the need for reading IN 127 and fiddling with that to get it to work with your printer and/or TS2068.

There is a test line at 1000 that I used to debug the routine. You could take that out.

Now a word or so about my new printer: it's a Fujitsu DL900 (sounds like an imported sports car!). This came in at about \$300, has a variety of fonts, prints in Portrait or Landscape mode, ie, the former on 8.5 x 11 sheets of paper, the latter with the same size paper turned sideways. This is handy for printing spreadsheets whose width exceeds the limits of the portrait mode.

R.H.MITCHELL  
20 WILD BRIARWAY  
WILLOWDALE ONT  
M2J 2L2  
JAN 02 1992

PRINTER INTERFACES AND DRIVER SOFTWARE  
A Primer by George Chambers - Part 3 - Aerco

In the last issue of this series we covered the Tasman printer interface and it's software. This article will continue the series with a discussion of the Aerco i/f and software.

The Aerco printer interface is perhaps the standard for the TS2068, a benchmark against which all others are compared. Certianly it is the most widely used one. As a measure of it's success maybe one could say the most-copied one.

The Aerco i/f is designed to deliver data to the parallel port of a printer. That is to say, it does not work to a serial printer port. The majority of printers come with a parallel port; some printers offer both serial and parallel ports, and a very few have serial ports only. The TS2068 buff would probably be wise to shun printers that do not have a parallel port.

The Aerco plugs into the rear connector of the 2068, and comes with a ribbon cable terminated with a standard Centronics-type 26-pin D-connector.

The software code used with the Aerco consists of 1111 bytes of code, located at starting address 64256. It comes with a Basic program which is used to customise the driver code to the user's printer specifications. This Basic program contains all the information needed to customise and test the software.

Rather than get into a dissertation on how the system operates I shall print out the instructions that come in the Basic program. They are as follows:

\*\*\*\*\*

CENTRONICS PRINT DRIVER  
is now loaded.

\* \* \* \* \* PRESS a KEY \* \* \* \* \*

C Customize

E Exit to BASIC

L Load your program from tape

N Notes on operation

P Printer demonstration

S Save on tape

The PRINT driver has 2 basic modes. The default mode is for use with the TIMEX BASIC system and automatically expands all tokens as they are encountered.

Select this mode by  
POKE 64256,1

The other mode is LITERAL where characters are sent directly to the printer with no changes made to them. Use the LITERAL mode to send control characters to the printer and for bit mapped graphics.

Select this mode by  
POKE 64256,0

To set the width of the printer:  
POKE 64259,width-1

To send LINE FEED after CARRIAGE  
RETURN: POKE 64260,10

To suppress LINE FEED after  
CARRIAGE RETURN: POKE 64260,0

To select Timex 2040 printer:  
POKE 26703,0; POKE 26704,5

To select Centronics printer:  
POKE 26703,5; POKE 26704,251

In order to COPY the screen,  
use the command LPRINT CHR\$ 1.

Don't forget to change all  
references in your programs  
from COPY to LPRINT CHR\$ 1

PLEASE SELECT YOUR PRINTER

Gorrilla Bananna . . . . .	1
Seikosha GP100 . . . . .	1
Prowriter 8510 . . . . .	2
Star Gemini 10X . . . . .	3
Epson (All older models) . . .	3
Epson (All newer models) . . .	4
Mannesmann Tally Spirit 80 . .	4
Olivetti PR 2300 (Zoom) . . .	5
Olivetti PR 2300 (Norm) . . .	6
Seikosha GP250X . . . . .	7

If your printer is not on this  
list, try the various types.

Contact AERCO at Box 18093  
Austin, TX 78760 for assistance.

Don't forget to SAVE your  
customized version on a  
DIFFERENT tape. (S on main menu)

This program needs to be told  
what kind of printer you have  
in order to properly COPY the  
screen.

If you do not have a dot matrix  
printer with BIT MAPPED GRAPHIC  
capability, you cannot COPY the  
screen. LPRINT and LLIST should  
still work fine on your printer.

To COPY the screen to the  
Centronics printer:  
LPRINT CHR\$ 1

Note that the details of the  
COPY command are different for  
various brands of printers.  
This package needs to be  
customized for your printer.

Please press the C key from the  
main menu to customize the  
software for your printer.

\*\*\*\*\*

There are four listings in this article. LISTING 1 and LISTING 2 demonstrate the use of the "CHR\$" and "OUT" commands to send control codes to the printer. Both listings are designed to do the same task, namely to produce the output shown in the Sample Printout. LISTING 4 is a Basic program to copy screen text to a large printer, while LISTING 4 is used to determine the "Printer Ready" indicator put out by the printer.

First we shall review LISTING 1 and 2. Lines 100 to 140 are the same in both listings. Their purpose is to condition the driver for use with a large printer. The REM statements describe the purpose of each line.

Both listings make use of three printer control codes: Chr\$ "M" or "77" for the Elite mode; CHR\$ "P" or "80" for the Pica mode; and CHR\$ "-" or "45", plus CHR\$ "1" and "0" to control the underline function. We have confined the demonstrations to these three values because they are common to most printers. There may be some printers that have a different set of printer codes, in which case you will need to modify the program accordingly.

#### The OUT Command

Now let us look at LISTING 1. This program makes use of the OUT command (on key "0"). To send a printer control code to the large printer via the Aerco Interface we use the command OUT 127,x, where x is the desired control code. Look at Line 200. Here we have sent two printer control codes, i.e. 27 and 77. Let's look at the first code, the 27.

The printer anticipates that all the data coming from the computer is printable characters. So that whenever we wish to send control codes we have to warn the printer that the next character (or characters) it receives are going to be control codes. We do that by sending the code 27. Whenever the printer sees that code in the data stream it responds accordingly.

In line 200 the code after the "27" is "77". In the majority of printers this means to switch to the Elite font. Simialrly, if it had received an "80" after the "27" it would know that the Pica font was to be used.

Now, one can send several successive codes following the initial "27". Look at Lines 260 and 300. Here we have two characters following the "27". Together, "45" and "49" tell the printer to go into the underline mode, while "45" and "48" remove the underline. One could send a longer string of codes, if the printer requires it, though this is not common.

One could also have added an OUT 127,10 to any of these lines, to give a LineFeed command. In Listing 1 we have put this instruction in a separate line 265, but it could have as easily been put at the end of Line 260. In Line 240 we have used a different command to get a linefeed. Either way is suitable.

(It's not clear to me how the printer recognises the end of a control code sequence, sorry!)

When using the OUT 127,x command we have to be careful. The computer can output control codes faster than the printer can act upon them. Therefore we must introduce a GOSUB routine (at Line 1000) to hold the program in a loop until the printer advises the computer it is ready to accept another instruction. One could use a PAUSE 5, or some other value instead of the GOSUB routine. Either method is acceptable, however the GOSUB routine is more reliable; the PAUSE interval might happen to be too short. Also, the GOSUB routine will probably be faster overall because of the cautionary length of the PAUSE value.

In the example given we have used a test value of 108 in Line 1000. This may not be suitable for your printer. To test it out, write a short routine to test for your printer's "ready" value. A typical routine would be similar to LISTING 4. Run the routine, make the printer "ready" and see what reading is produced. Use that value in Line 1000. If you are curious, try the printer in "off", "no paper", and any other states it may have.

You may note that, in Line 130, we have placed the driver in what is termed the LITERAL mode. We can leave it in that mode for all purposes, except if we wish to expand the Sincalir tokens such as when we wish to do an LLIST. Look at line 350, and you will see that we have changed to the TIMEX Basic mode so that an LLIST will come out properly. Try LLIST in the LITERAL mode, and see what happens!

#### The CHR\$ Command:

We have made use of the CHR\$ command in LISTING 2. The first thing you may notice is the absence of a GOSUB routine that was used in Listing 1. This is a feature of the CHR\$ command; the computer does not send out another instruction until the printer acknowledges receipt of the current one.

In this listing I have varied the make-up of several lines to demonstrate possible programming variations. Compare lines 260 and 300. They look different, however except for the fact that one turns the underline on (the "1"), and the other turns it off (the "0"), both lines are effectively the same.



Note the use of "LRINT CHR\$ 10" in several lines. This gives a Line Feed. The linefeed commands are a bit tricky to grasp. Use the command 'LRINT CHR\$ 10', and you will always advance the line feed immediately. However the command LPRINT will not produce any movement. Similarly, Entering the command 'OUT 127,10 will result in no visible action. But if you enter the 'OUT 127,10, and follow it with 'LPRINT" the line feed will advance. A subsequent LPRINT command will fail to move the line-feed however; unless it is prefaced by another 'OUT 127,10' instruction. Experiment with this until you are familiar with the principle.

The Aerco documentation mentions that you can make a screen (graphics) copy by using the command 'LPRINT CHR\$ 1'. This may work on some printers; it does not on my SCM Fastext 80 printer. I think it has to do with the control codes used to set the printer in the graphics mode. AERCO, in their literature, suggest that they be consulted in the event of problems. Probably the feature needs to be customized to particular printers.

### LAWS OF COMPUTER PROGRAMMING

*from the AERCO instruction tape*

*Any given program, when running,  
is obsolete.*

*If a program is useless, it will  
have to be documented.*

*If a program is useful, it will  
have to be changed.*

*Any program will expand to fill  
all available memory.*

*The value of a program is propor-  
tional to the weight of it's ou-  
tput.*

*Program complexity grows until i-  
t exceeds the capability of the  
programmer to maintain it.*

*Make it possible for programmers  
to write in English and you wi-  
ll find that programmers cannot  
write in English.*

*If builders built buildings the  
way programmers wrote programs,  
then the first woodpecker that  
came along would destroy civil  
ization.*

*Inside every large program is a  
small program struggling to get  
out.*

*The attention span of a computer  
is only as long as its power c-  
ord.*

*If a test installation functions  
perfectly, all subsequent syst-  
ems will malfunction.*

*Not until a program has been in  
production for at least six mon-  
ths will the most harmful error  
be discovered.*

*One good reason that computers c-  
an do more work than people is  
that they never have to stop an-  
d answer the telephone.*

*If you put tomfoolery in a compu-  
ter nothing comes out but tomfo-  
olery. But this tomfoolery, hav-  
ing passed through a very expen-  
sive machine, is somehow enoble-  
d and none dare criticize it.*

### LISTING 1

```
100 POKE 26703,5
110 POKE 26704,251: REM Points
the 2068 to the Aerco Printer
driver code.

120 POKE 64259,31: REM Sets
line length to 32 characters.

130 POKE 64260,10: REM To give
LineFeed with Carriage Return

140 POKE 64256,0: REM To put
code into the LITERAL mode;i.e.
to output printer control codes
.
150 LET m=1000
200 OUT 127,27: GO SUB m: OUT 1
27,77: GO SUB m
210 LPRINT "This is Elite"
230 OUT 127,27: GO SUB m: OUT 1
27,80: GO SUB m
240 LPRINT CHR$ 10
250 LPRINT "This is Pica"
260 OUT 127,27: GO SUB m: OUT 1
27,45: GO SUB m: OUT 127,49: GO
SUB m
265 OUT 127,10
270 LPRINT "This is underlined
Pica"
280 OUT 127,27: GO SUB m: OUT 1
27,77: GO SUB m
285 OUT 127,10
290 LPRINT "This is underlined
Elite"
300 OUT 127,27: GO SUB m: OUT 1
27,45: GO SUB m: OUT 127,48: GO
SUB m
310 LPRINT CHR$ 10

350 POKE 64256,1
360 LLIST
380 STOP
1000 IF IN 127<>108 THEN GO TO
1000
1010 RETURN
```

### LISTING 3

100 REM Listing used to copy  
screen text to a  
large printer

```
1000 FOR n=0 TO 22
1010 FOR m=0 TO 31
1020 LPRINT SCREEN$ (n,m);
1030 NEXT m
1040 LPRINT
1050 NEXT n
1060 STOP
```

### LISTING 2

```
100 POKE 26703,5
110 POKE 26704,251: REM Points
the 2068 to the Aerco Printer
driver code.

120 POKE 64259,31: REM Sets
line length to 32 characters.

130 POKE 64260,10: REM To give
LineFeed with Carriage Return

140 POKE 64256,0: REM To put
code into the LITERAL mode;i.e.
to output printer control codes
.
200 LPRINT CHR$ 27;CHR$ 77
210 LPRINT "This is Elite"
230 LPRINT CHR$ 27;CHR$ 80
240 LPRINT CHR$ 10
250 LPRINT "This is Pica"
260 LPRINT CHR$ 27;CHR$ 45;"1";
CHR$ 10
270 LPRINT "This is underlined
Pica"
280 LPRINT CHR$ 27;CHR$ 77;CHR$
10
290 LPRINT "This is underlined
Elite"
300 LPRINT CHR$ 27;"-0"
310 LPRINT CHR$ 10

350 POKE 64256,1
360 LLIST
380 STOP
```

### LISTING 4

```
10 PRINT AT 10,10;" ";AT 10
,10;IN 127
20 PAUSE 20: GO TO 10
```

### Sample Printout Listings 1 and 2

```
This is Elite
This is Pica
This is underlined Pica
This is underlined Elite
```

# - - - N E T W O R K - - -

## GOLD CARD AND JSU ROM NOT COMPATIBLE

by Hugh Howie

At the beginning of July I sent my order for the Gold Card to Miracle Systems. It arrived in the middle of August so I very quickly got it into service, and was it ever quick. I could hardly believe how fast it was, and all that extra desk space compared with my Expanderam and Cumana interface. But you have heard all this before.

The first serious test came when I tried to print out a text87 document in two columns. No matter what I did the two columns ran together, and the typeface and spacing could, and did, change in the middle of the line. I replaced the Gold with my original combination of Cumana and Expanderam and the document printed out fine.

Switched to my #2 QL with Trump Card and once again the print out was OK. Back to the Gold and back to the problem again.

Sent a letter and samples off to Miracle systems, and in a couple of weeks I got a nice letter from Mike Tomlinson of Miracle, and also a replacement ROM which he said would fix the problem, and it did. Something to do with a time dependency in the Gold Card. The printout was all that could be desired.

The ROM he sent was a 2.22, to replace the 2.15 which came with the Gold.

Later I tried to Network, and could not get anywhere. Up till now I had been able to network with a not unreasonable amount of success. Many tests were carried out using the Gold Card. I went to my #2 machine with Trump and Minerva and replaced the Trump Card with the Gold. The Trump going to #1. Network was now working.

Head scratching was in order as you can imagine. Another letter off to Miracle, and some more tests and much switching of Gold and Minerva. I also was able to use four QL's in various combinations, and the only time I could network was when the Gold Card was in the same machine as Minerva, which uses the JSL1 ROM. The Gold Card in the same QL as the JSU was a bust.

I sent the results of those test off too Miracle in October 21/91, and I am awaiting their reply. Miracle Systems have been more than helpful in a couple other problems I have had, but those particular problems do not belong here.

I again wrote to them on November 22nd, as a follow up to my previous letter.

I understand that there is a 2.25 ROM available, but I have no knowledge if it has cured the network problem.

To sum up. The Gold Card is fast. To be able to use discs with 2880 sectors is nice. But if you want to Network with the Gold. HOLD OFF in the meantime. I have asked Miracle if the Gold Card will work with the MGUK ROM.

I don't relish the thought of changing my ROMs to suit the Gold Priced Gold Card, an add on that costs more than ALL my QL's put to-gether!

So, if you are interested or involved in Networking, and are contemplating acquiring the Gold Card, don't rush out and buy it just yet. I will let you know if and when my own Network problems are solved. In the meantime.....

### H O L D   O F F

SINC-LINK

December 21/91

323 1/2 N. Church Street  
Bowling Green, OH 43402  
November 4, 1991

Dear Bob,

First, I do know about things like putting a space instead of a blank REM and two different ways to get a "PRINT comma". I don't usually use them in things I am writing for other people, especially if they will be sent on paper, since I am not certain if all the people I write to would recognize those and press the same things. Obviously you do, but I'm not sure if the newsletter readers would, and might wonder when their program didn't look the same as the listing. Though there are differences in the listing anyway, since the printer driver handles tokens differently from the way they appear on the screen.

As far as MS-DOS goes, BBSs and direct connections are the obvious approach. That's right, you should be able to phone someone's IBM directly. Also if you had a serial interface, you could set them up that way.

It is theoretically possible to write a program that could read and write MS-DOS disks, but it is not a minor task. One problem is that LKDOS writes a full track each time, while MS-DOS writes a single block (two sectors, or 1/5 of a track). I have all the information on file formats and directory setup, but the difficulties involved have kept me from writing such a program (yet). Of course, someone could also write a program on a PC that would read LKDOS disks, but you would have to find someone who knew ml on the PC and the LKDOS format.

Oh, there is one item I did on the clock that I might like to change. The set-up routine only sets up the interrupt table. Yours also displays the initial clock setting. So on mine, you end up waiting 1 second for the clock to appear. I should also have made the clock itself set the attributes where it is displayed. That way, if you mistakenly print something there (or LIST) with the same paper and ink, the clock would still be visible the next time it updated. I've seen some programs with embedded INK and PAPER to make them unreadable. Of course, any other program that diverts the interrupts will disable the clock.

I note in passing that the original clock on the IBM had to be reset whenever the machine was turned on, but all the current models seem to include a battery-backed clock of some kind.

Just for your information, I have seen programs on other machines that let you run multiple programs at once. We could do this too, we would just have to save the stack and system variables (and the registers, of course). In fact, true multi-tasking systems do work on other computers, but they don't need to worry about which ROM they are seeing. We have (at least) 3 ROMs to worry about - the HOME ROM (with most of BASIC in it), the EXROM (with the cassette routines) and LKDOS.

If you wish to find out all about MS-DOS disks, I would refer you to BYTE magazine, June 1989, page 327 and following. I could copy out the appropriate material, if you wish. I may get back to writing a program to handle all that, but as I said

it won't be simple.

Heres a little known piece of trivia for you. Did you know that how fast an ml program runs is related to what address it resides at? That is a consequence of some information buried in the technical manual for the 2068. If you put a program between 4000h and 7FFFh, the CPU has to compete with the video chip for time, whereas a program between 8000h and FFFFh can be accessed by the CPU while the video chip access the screen. How much of a difference does it make? Well, I just wrote a short program to count to 65576 in ml, and timed it at 3 different locations. When I put the code in the printer buffer, it took 25/60 seconds. When I put the code at 33000 or 65300, it only took 18/60 seconds, a reduction of about 30%! Of course, if your program does something that requires waiting (printing on the printer or screen, or disk access), a move like that won't make a difference. But if you do write something like a sort or whatever that is actually working all the time it is running, putting it above 32768 (or possibly in the cartridge RAM) instead of between 16384 and 32767 will save you time.

I'll include a listing and output of the program I used to test it. The "Time: " numbers are in 1/60ths of a second (frames) - that is the most accurate independent timer I had. Since the ml both clears and reads the system variable FRAMES, there is no BASIC overhead to worry about, and my numbers are more accurate than doing the POKES and PEEKs in BASIC would be. Oh, I didn't test it, but pressing a key may change the times anywhere you locate your code.

I have almost finished the MScript file compressor I promised you some time ago. I'll certainly include that next time. Have a Happy (American) Thanksgiving, and I'll write soon. Peace!

Sincerely,  
*Steven V. Gombour*

P.S. I stand corrected with regard to pressing a key. I got the same 25, 18, and 18 with any key pressed as without.

I have also replaced my printer ribbon (again). The cheap ribbon I used last time was too light, and didn't last very long. This should be better (compare this letter to the program listing for the speed test).

*P.P.S. I did start this 11/4, as the heading says, but finished it 11/20.*

```

1 REM program- speed test
2 LET code=100
10 LET start=23300: GO SUB code
20 LET start=33000: GO SUB code
30 LET start=65300: GO SUB code
40 STOP
100 RESTORE 1000: FOR i=0 TO 17: READ a: POKE start+i,a:
NEXT i
110 PRINT "start: ";start,"Time: ";USR start
120 RETURN
1000 DATA 1,0,0,237,67,120,92,13,32,253,
5,32,250,237,75,120,92,201

start: 23300   Time: 25
start: 33000   Time: 18
start: 65300   Time: 18

```

retyped. Rm.

#### QL TIT-BITS

##### MINERVA ==

Some time ago some of us purchased, Minerva, to replace the ROM in our QL's, the advantage was that some of the bugs of the original ROM were corrected. Also some games for the original ROM would not function properly with added memory, such as TRUMP CARD or even CUMANA interface. The graphics required 128 K ROM only and then you would have to use SHRINK or RES128 to fool the QL into thinking that only 128 K of ROM was available. With MINERVA, on the original display you have the option of using only 128 K of memory by using " SHIFT " with "F1 or F2" It saves a lot of bother !!!!

##### text 87

Not much more to report except that it is proving quite a worthwhile word processor, and the learning process is rather extensive.

##### LAN

##### Local Area Network :

After the demonstration last year of LAN, I got interested in setting up a network, because it did not involve any expense, since when I purchased TRUMP CARD, in addition to the extra memory, I also got TOOLKIT II built in. With Cumana interface, TOOLKIT II plug-in ROM, and my back-up QL I had everything needed. As a matter of fact my two drives 3.5 " connected to my regular QL " NET 1 " and my two 5.25" drives on the " NET 2 " machine I was ready to experiment. There is not much information on NETWORKING but a few articles in UPDATE during 1989 and 1990 as well as a lot of help from Hugh Howie I was able to communicate between NET 1 and NET 2 and copy files from n2\_flp1\_ ( 5.25 " ) to n1\_flp1\_ ( 3.5 " ) which is what I wanted to do in the first place.

=====

*Louis Lafreniere*

This is a short program which enables a Basic instruction to be translated into machine code without the use of an interpreter or assembly code. When you have finished entering it, simply press RUN and all will be made clear.

```

5 REM FOR ZX81 IN SLOW MODE
10 LET A=16509
15 PRINT TAB 2;PEEK A*256+PEEK
(A+1)
20 LET L=PEEK (A+2)+256*PEEK (
A+3)-1
25 LET A=A+3
30 FOR N=1 TO L
35 IF PEEK (N+A)<>126 THEN GO
TO 50
40 LET N=N+5
45 GO TO 55
50 PRINT CHR$ (PEEK (N+A));
55 NEXT N
60 LET A=A+N+1
65 IF A<16914 THEN GO TO 15
70 LET A$="-9825 .885 "
75 FOR N=1 TO LEN A$
80 FOR J=1 TO 50
85 NEXT J
90 PRINT AT 15,12+N;CHR$ (CODE
A$(N)+144)
95 NEXT N

```

Taken from the April '82 issue  
of Your Computer - Vol.2 No.4

While messing around on my 16K ZX-81 I found a way to make the screen Clear quickly after scrolling. Normally a CLS command or a return to non-Scrolled Printing takes a long time because the display has to be padded out with spaces on the expanded machines. This program will illustrate this:

```

10 FOR N = 1 TO 22
20 SCROLL
30 PRINT "TEST"
40 NEXT N
50 CLS

```

See how long the CLS command takes. The trick is to artificially pad the display file on each scroll, by using a Tab to move the Print position to the end of the line. To show this, change the line 30 in the program to:

```
30 PRINT "TEST"; TAB 31;
```

and RUN the program again. By forcing the ZX-81 to print a full line of 32 spaces each time, the display file remains intact.

Taken from the November 1982 issue  
of Your Computer...Vol.2 No.11

## Put some punch in your PRINT FACTORY® graphics!!

Collection#1	Collection#2	Collection#3
Miscellaneous	Animals	Digitized Misc.
Men/Boys	Funny	Female Faces
Women/Girls	Transportation	Male Faces
Digitized Misc.	Sports	Funny
	Computers	Animals
	Holidays	Zodiac Signs
	Food	Holidays
	Xmas	Transportation
		Xmas



Over 150 different large & small  
graphics in each collection!

\$12.95 ppd each, \$19.95 ppd for two, or  
all three collections for \$26.95 ppd.  
[Check or money order only.]  
Available on tape or 5 1/4" DSDD LARKEN  
or OLIGER disk - please specify.



Send order to:

John McMichael, 1710 Palmer Dr.,  
Laramie, WY 82070



# ONE SIDED DISKS

## Open letter to Gil Parrish and Steve Gunhouse

Gentlemen,

Your separate replies to my "One Sided Disc" article was entirely what I had hoped to see. I was delighted to learn that someone took the trouble to read my 'writings'. (for sake of a better word to use) Both of you had excellent comments to make, and I found the comments most interesting.

There is considerable ignorance on this subject, and as it is a subject with which most of us have to deal at one time or another, I would like to offer an invitation to one or both of you to write something expanding on what has already appeared in this Newsletter. I feel that such an article would be appreciated by all our readers.

My comments were merely a lead-in to show that anyone could format a disc to be one-sided disc, and to get such knowledgeable feedback was a most unexpected pleasure.

Please ?



### TS2068 - TIMACHINE - A FIX

Some time ago one of our members, Bob Mitchell, identified a shortcoming in the Timachine compiler. The problem was described as follows:

"Printout of Timachine Runtimes different when sent to large printer than when sent to screen. Onscreen, (with my program) approximately 20 Runtimes were printed; on paper, about twice that many. Also, Variable List on paper not the same as onscreen. It seemed to be incomplete on paper, and corrupted."

I wrote to the author of TIMACHINE, Cameron Hayne, who happens to be a former member of our club. I asked whether he had an answer to this problem. Cameron replied, saying that his 2068 was packed away because of a house move, but suggested it could be due to the use of a PRINT AT instruction in TIMACHINE that was inappropriate when using a large printer. He also suggested places in the TIMACHINE coding to look for the problem.

Presently, I asked another club member, Ken Shoenberger, if he would take a look at it. Ken did so, and came up with a modification which corrects the problem. I asked Ken for a write-up for the newsletter. It is elsewhere in this newsletter.

If any club members having Timachine and a large printer, wish to get a copy of this revised version, Simply ask me. State whether you need it on tape or Larken disk.

George Chambers



## TIMACHINE - A Fix by Ken Shoenberger

The purpose of this write-up is to explain what was done and why to enable large printer printing of the LIST report after compiling a program with TIMACHINE.

The first obstacle was getting the large printer to print anything. LPRINT after opening #3 to "lp" only gave a dot and two dashes on the screen at about line 10. This was traced to the relocation of the Channels but not changing them when TIMACHINE initialized the location of any BASIC program. Unlike the TIMACHINE for SPECTRUM, the start-up for LKDOS in 2068 mode modifies the channels above "P" to go to locations just above the Channel addresses. When the location of the channel addresses were moved they still pointed to locations that were now occupied by the MC for TIMACHINE. These were changed to point correctly and the length of the MC was increased to include them in the SAVE instruction.

The next problem was with the printout. In TIMACHINE this report is tabulated by "print at" instructions which the large printer ignored. "Print a space" was substituted for all "print at" instructions except the ones that put the printing at column zero.

The next problem was with the Runtime numbers and the addresses of their locations. This was traced to the use of the IX register, which pointed to the location of the Runtime addresses in the Display File. Each time the program sent the operation to print anything, LKDOS would reset the IX register for its use and the IX register would be corrupt when TIMACHINE wanted it the way it had left it. What was necessary was PUSH IX before going to LKDOS to print and POP IX before using it for TIMACHINE use. This was done by moving and eliminating instructions in the LIST portion of the TIMACHINE program.

The printing of the simple variables was all right and nothing was needed here except dropping the "print at" in favor of "print a space" which netted three bytes to use for the "PUSHes" and "POPps" that had to be added.

The array variables and strings also required PUSH IX and POP IX additions to get around the corrupt pointer for address locations.

All the foregoing was done using debugger DB40XX.C2 program starting with disassembly of "TSTI2.C1," previous disassembly of LKDOS and ending with the moving and modifying the LIST portion of "TSTI2.C1." The modified program is called "COMPIL.C1" and the BASIC loader is "TSTME.B2."

TS2068-Larken NMI-F key  
Another Application

While incorporating an NMI-F key function in the OMNIBUS menu program (TTSUC Library Disk #2) I realized I had another application for it. This NMI-F function is one where by pressing the NMI- button, followed by the "F" key on the computer, an AUTOSTART load is initiated on the target drive.

I have a great many disks with Spectrum games on them. Each disk has maybe 8 games on it, plus an AUTOSTART menu program. I decided that it would be worthwhile to include the same NMI-F function to these disks. The advantage would be a quick easy return to the menu. The following describes the procedure I used to install the feature.

First, I entered and ran the Basic program in Figure A. Running the program placed a block of code, starting at address 23310. Figure B is a disassembly of the code. Using the command 'SAVE "AUTOSTART" CODE 23310,60', I saved the code to tape. Note that I saved it to tape, not disk. There's a good reason for this; it makes the task simpler!

I then loaded the AUTOSTART menu program that I wished to modify. Near the beginning of the program I entered a new line, as follows: '9 PRINT #4: POKE 8200, 16350: RANDOMIZE USR 23310 '. Next I loaded back into the computer the code saved earlier from tape. And finally I re-saved the AUTOSTART menu by means of a line in the program: '3 CLEAR 26000: RANDOMIZE USR 102'.

Now, a couple of further tips. Because I had a great many disks with the same menu format, I modified my technique somewhat. I saved the first 9 lines of the modified menu program to tape. See Figure C.

I saved the 9 lines to the same tape, immediately following the code saved earlier. With each successive disk, all I had to do then, was load the AUTOSTART menu; enter as a direct command 'LOAD "" CODE: MERGE ""', and play the tape. The code loaded, and the Basic overwrote the first nine

lines of the original program. I then simply did a GOTO 3 to save the now modified AUTOSTART menu.

One can test that the job has been done correctly by doing an NMI-F operation after the SAVE; the disk should reload the AUTOSTART menu.

Now, This code as listed in Figure A, directs the Larken DOS to look at Drive 0, the first drive on the system. If you wish to designate another drive, the solution is to change one of the numbers in the listing in Figure A. In line 32, replace the number "2". Use Figure D to select the proper value for the desired drive.

George Chambers

FIGURE A

```
10 FOR n=23310 TO 23369
20 READ a: POKE n,a: NEXT n: S
TOP
30 DATA 243,205,98,0,33,36,91,
17,222,63
31 DATA 237,83,22,32,1,34,0,23
7,176,195
32 DATA 186,0,62,2,50,3,32,33,
246,63
33 DATA 17,34,32,1,10,0,237,17
6,62,1
34 DATA 50,2,32,195,198,0,65,8
5,84,79
35 DATA 83,84,65,82,84,32,0,0,
0,0
```

FIGURE C

```
1 REM          DISK MENU
2 GO TO 4
3 CLEAR 26000: BORDER 0: PAPE
R 0: INK 0: CLS : RANDOMIZE USR
102
4 CLEAR 65366: PRINT USR 100:
OPEN #4,"dd"
5 BORDER 1: INK 7: PAPER 0: C
LS
6 PRINT AT 1,2; PAPER 2;"
GAMES DISK MENU      ""
8 RESTORE 100: POKE 23658,8:
LET x=8
9 PRINT #4: POKE 8200,16350:
RANDOMIZE USR 23310
```

cont.

# FIGURE B

23310	F3	DI	
23311	CD6200	CALL	98
23314	21245B	LD	HL,23332
23317	11DE3F	LD	DE,16350
23320	ED531620	LD	(8214),DE
23324	012200	LD	BC,34
23327	EDB0	LDIR	
23329	C3BA00	JP	186
23332	3E02	LD	A,2
23334	320320	LD	(8195),A
23337	21F63F	LD	HL,16374
23340	112220	LD	DE,8226
23343	010A00	LD	BC,10
23346	EDB0	LDIR	
23348	3E01	LD	A,1
23350	320220	LD	(8194),A
23353	C3C600	JP	198
23356	65	A	
23357	85	U	
23358	84	↑	
23359	79	O	drive 0= 2
23360	83	S	drive 1= 4
23361	84	T	drive 2= 8
23362	65	A	drive 3= 16
23363	82	R	drive 4= 128
23364	84	T	

# FIGURE D

drive 0= 2  
drive 1= 4  
drive 2= 8  
drive 3= 16  
drive 4= 128

**GOLD**  
**IS**  
**BAD BUY**  
**MORE PROBLEMS DEVELOP**  
**will NOT format MDV with JSU**  
**IS OK WITH MINERVA**  
**? HOW MANY MORE PROBLEMS ?**  
**NEXT EPISODE - NEXT ISSUE**

# Q L I P S

By Hugh H Howie

## QUICKIES.....

Here is another of those QUICKIES I like to come up with now and again.

Last issue I mentioned the WSTAT command from TK2. Well here is a little to amplify that command. Too often we want to know the amount of occupied sectors on a disc, and we type in DIR xxxx, and get a list of everything which is on that disc. Next time this happens to you, just type in :-

STAT <ENTER>

That's it. You will be shown the number of free sectors on the disc and NO LONG LIST. Very Quick!.

Have you got a shortie? Lets see it.

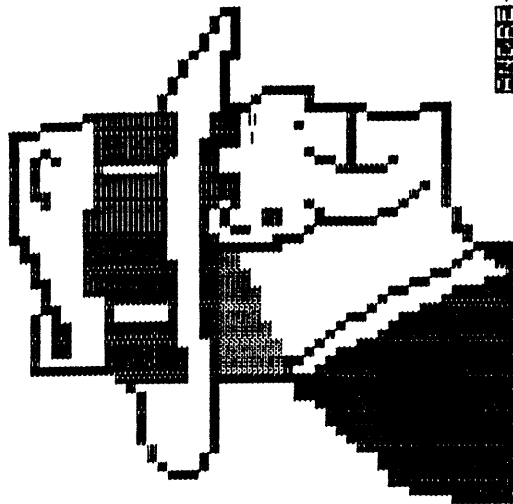
## A NEW NEWSLETTER

Recently we received a newsletter devoted to the ZX-81 computer. It consists of two pages. The Editor, Andre Baune, of Chateauguay, Quebec, promises to continue the newsletter for at least a year, depending on the interest and feedback he receives. He mentions that he has sent out 227 copies. Gosh, I didn't know there were that many Timex enthusiasts around, let alone ZX81 fans!!

We have included his newsletter in this issue. Hope you like it. Drop Andre a line; the address is in his newsletter.

George Chambers

\*ALL\* MADE ON ZX81/TIMEX 1000



\*\*\*\*\*

Send the name of this character and you will receive the program and the DATA that will put this picture on your own screen (or printer). See bottom of page 2 for more details.

TO: 14 RICHMOND COURT  
SCARBOROUGH, ONTARIO

MIK 2YI

TORONTO TIMEX SINCLAIR USER CLUB

# ZX-81

## 10 YEARS LATER

ALL

THE NEW NEWSLETTER FOR THE  
SINCLAIR ZX81 / TIMEX 1000.

ALL YOU SEE, AND WILL SEE, HAS  
BEEN CREATED WITH, AND ONLY, A  
ZX81, A 16K RAMPACK AND A TIMEX  
2040 PRINTER.

### WHY ZX-81?

Because the ZX81/Timex 1000 is 10  
years old this year and I want to lift the  
spirit of the users and the believers in  
this marvelous little computer. I also  
want to reach the far away users and/or  
programmers.

In this newsletter you will get  
print-out, pictures, and articles like  
X-TRA BASIC, ALL ABOUT PAUSING, ALL ABOUT  
PRINT, BYTE SAVING, FASTER RUNNING  
PROGRAMS, SPEED TEST, PROGRAMMER'S TOOLS,  
TIPS 'n TRICKS, etc...

→ page 6

### WHO IS ANDRE\*\*\*

My name is Andre Baune and I am  
forty-nine years old but I still move,  
think and act like a sixteen years old.

I bought my first Sinclair computer in  
December 1981. It was my personnel gift  
for Christmas. I had no previous  
experience or knowledge about computer but  
what I had seen and heard about them had  
made them dream machines. They had become  
amazing, fantastic, interesting, baffling,  
out of this world. For the first time man  
had created a machine that mimicks his own  
brain and now it was available to  
everybody for a cheap price. WOW! And I  
could get one!

I got one! When I opened the box and I  
saw that little flat black thing, my heart  
sank. No way! That thing cannot perform. I  
connected it, started to read the  
instructions and tried it. It worked! It  
worked more, much more than I thought.

For me, learning being fun, after two  
months I was able to create programs and  
as a reward I allowed myself to connect  
the 16K ram module. Another two months  
passed and I bought my first program.  
Since then I did not buy many programs but  
I created, adapted and modified hundreds  
of programs and I still do create and  
learn more and more about and with my  
ZX81.

The best part of this story is that I  
AM STILL LEARNING and through this  
newsletter I will communicate to you what  
I have learnt through the past years.

All you will see in these pages has  
been done in BASIC because I did not start  
yet to learn machine language. So if you  
are just a beginner it should be easy to  
learn and understand my explanations about  
BASIC language because what I talk about  
is the BASIC language of your ZX81/Timex  
1000. If you are a more experienced  
user/programmer there will be new methods  
or tricks to make your computer run better  
and faster without changing any hardware.  
So bring your Sinclair/Timex back on your  
working table, there is something new for  
them. This is what ANDRE\*\*\* is all about.

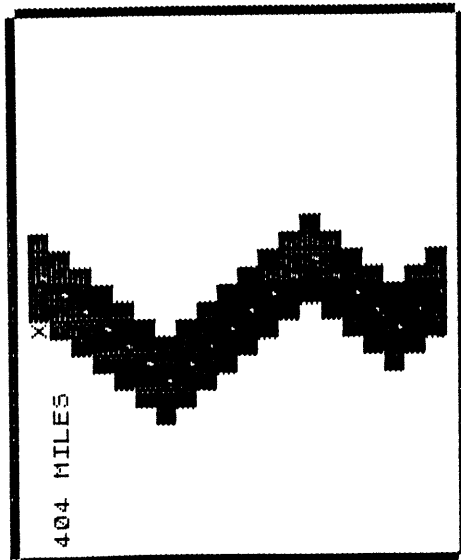
# ANDRE\*\*\*

IF YOU WISH TO RECEIVE  
A PRINT-OUT, A CASSETTE,  
AN INFORMATION, A REPLY  
OR THE NEXT MONTH ISSUE  
OF THIS NEWSLETTER SEND  
A SELF-ADDRESSED ENVELOPE

TO: ANDRE BAUNE  
304 SCOTT  
CHATEAUGUAY, QUEBEC  
CANADA J6J 4H5

## ZX CAR RACE

A SCREEN SAMPLE:



THE PRINT-OUT:

```

1 FOR I=A TO RAND*5+A
2 PRINT AT 0,C;" ";AT 21,R;"■"
3 LET M=M+A
4 LET R=R+T
5 LET C=C+(INKEY$="0")-(INKEY$="1")
6 SCROLL
7 IF NOT PEEK (D+C) THEN GOTO 11
8 NEXT I
9 LET T=(RAND*.5)*2+(R(5)-(R)1
10 GOTO A
11 FOR I=A TO 9
12 PRINT AT 0,C;"X ■ X ■ X"(I)
13 NEXT I
14 PRINT M;" MILES"
15 PAUSE D
16 LET R=13
17 LET C=15
18 LET M=NOT PI
19 CLS
20 FOR I=M TO 20
21 PRINT TAB R;"■";
22 NEXT I
23 GOTO 9
24 SAVE "ZX RACE"
25 PI
26 LET A=SGN PI
27 LET D=VAL "PEEK 16396+256*P
28 LET D=VAL "PEEK 16397+A"
29 GOTO 13

```

## ABOUT ZX CAR RACE

This extremely short program (463 bytes) runs in only 2K of ram (the standard Tinx 1000) and uses ALL the screen. The keys to steer the car are 0 and 1, located on each side of the keyboard at the top for easy handling. After an accident ANY KEY pressed will restart the game. This program uses 2 functions of the Sinclair/Tinx: PAUSE and SCROLL. The screen picture is enhanced by the added white line in the middle of the road.

## TYPING NOTES:

-Line 2 "inverse H"  
 -Line 2 and 28 "inverse space, inverse space, inverse dot, inverse space, inverse space"  
 -Line 12 "X, space, inverse X, space, X, space, inverse X, space, X"

## USER'S NOTES:

-TO SAVE: type RUN 34.  
 -TO start: type RUN 36.

## VARIABLES USED:

A = 1  
 I = utility counter  
 M = Miles run.  
 C = Car position  
 R = Road position  
 D = Display file address  
 T = turns value

## SINCLAIR/TINEX SUPPORTERS

Computer Monthly,  
 P.O. Box 7062  
 Atlanta, GA 30357-0062  
 Mountainer Software,  
 749 Hill Street # 9,  
 Parkersburg, WV 2604

Send a letter to these people with your name and address, asking them what they have for you and your computer.

As I gather more supporters' names of the ZX81/TINEX 1000, I will include them in this column.

## NEXT MONTH

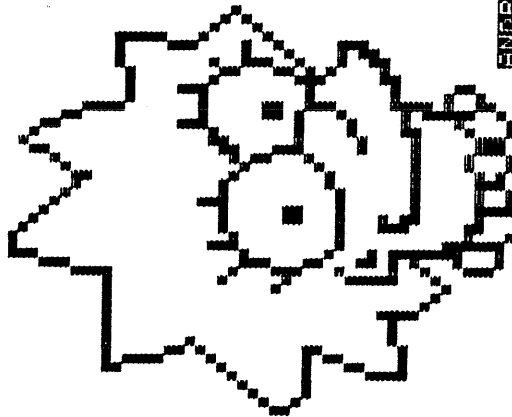
12 PAGES of more new GRAPHICS, NEW pictures, new print-out, PLUS new games, quizzes, PLUS new series of topics about OUR Sinclair/Tinx. SEND FOR IT→ NOW!

From page 1...

I will try to keep this newsletter for at least a year depending upon the interest and the feedback I will receive. So be nice and drop me a word, a name and address, a question, a suggestion. A problem, a small program you would like to be shared, a picture you would like me to digitize.

Join a club, send your name around and we will all gain because newcomers will join in and bring new ideas we all could benefit from.

HER ! ON SINCLAIR ZX81/TINEX 1000 ? ? ?



\*\*\*\*\*

Send the name of this character and you will receive the PROGRAM and the DATA that will put this picture on your own screen (or printer). See bottom of page 2 for more details.

## ACHTUNG

The content of the newsletter ZX-91 is the sole ownership of ANDRE\*\*\*. It may be reproduced in part or in whole providing that a copy of whatever it is reproduced in is sent to me and that credit is mentioned. ZX-91 is free and can not be used in any remunerative way.

! @ ~ ! + + + + + ! ! @ - [ \_ ] ~ % & \ #  
 The word processor used in these pages is "WORD SYNC II" of P. Hargrave, Nanaimo, B.C. Canada.

# SOME THINGS IT CAN DO

By Hugh Howie

Do you know all that this can do for you? I just bet you don't. Neither do I.

Recently I asked a few QL'ers what they used the Toolkit for, and I was not surprised at the replies I received. Most owners just used the wild card commands such as WCOPY and WDEL. I find that I use those commands as often as any other, and I know that Toolkit is necessary for Networking, at least it makes the Net easier. So I started to look at the manual, and although the print is small I found that there are a lot of useful things in there. More on the small print in the manual later.

If you have the Toolkit on Trump or Gold you have to initialise it by typing TK2\_EXT. But if you have it as a plug-in to the ROM port at the back of the QL, it initialises itself on power-up, which is really much handier and releases the user from having to remember to initialise it each time you reset or power-up. However you have it, it is the most important tool in your little black box of buttons.

The first nice thing is that all read/write commands are directed at flp1\_, so that it is no longer necessary to type in the flp1\_ in most commands. Instead of typing in "LOAD FLP1\_TITLE" you can get away with "LOAD TITLE". Of course for flp2\_ or anything else you still have to designate that drive.

By use of the ALT+ENTER key it will bring to the screen the last command typed in since the ENTER key was pressed. If the ENTER has not been pressed for a while, by repeated use of ALT/ENTER you can go back quite a few commands. As an experiment try typing:- one <ENTER> two <E> three <E> four <E> five <E>. Now press ALT/ENTER a couple of times. (Section 20 Console Driver) This is extremely helpful if you should happen to make an error in entering a command. Instead of re-entering the whole command, just press ALT/ENTER and the command comes back on screen where you can edit the wrong letter. Saves a lot of re-typing

XTRAS typed in will display all 120 (or so) extra commands that have been made available by Toolkit 2.

VIEW TITLE will put on screen the listing of "TITLE" or whatever. (Section 3.3 Viewing a file)

Just to see what can happen try these:-

PRINT DATAD\$

PRINT PROGDS

PRINT DESTDS

Now to finish off that little lot try:-

DLIST Thats right, just as it is, alone.  
(Section 4 Directory Control)

PRINT FREE\_MEM will tell you the amount of free memory remaining. WMON or WTV will switch you back and forth between the TV and MONITOR screens. In many cases this switch can be useful to assist in clearing up the screen instead of doing a reset.

WDEL will delete files on a disc as fast as you can press the Y key. Like all the wild card commands you have the choice of:- 'Y'es 'N'o 'A'll 'Q'uit

WSTAT will give you the size of each file in flp1, and also the date it was saved or copied. whichever is later. STAT will give stat of disc with no DIR.

There are so many things like that, even the renaming of files on disc, Sections 5.5.2 and 5.5.4 cover this nicely. You will have to play with this one to find out what it can do.

Did you know that with an EXEC file it is sometimes possible to step into/out of S/Basic? Instead of using EXEC just use EX. (Section 8.1 and 8.2 Single Program Execution) n

There is even an ALT/KEY in which you can define your own keys. (Section 20.1)

Disc with programs in flp1\_ try DIR <enter>

There is an ALARM clock in the Toolkit. (Section 2.3 #18 Time Keeping) Just to remind you that it is time to go to bed!

There are so many useful things in here. I know that most of you are aware of some of the goodies, and this little bit is just to remind you to take another look at the manual, and see what you are missing. So often when we pick up a few tips we are inclined to say thats enough, but there are so many little tools for so many little things, I would not be surprised if I were to find a tool to change a membrane quickly. Now that would really be something! All I am trying to do is get you to take another look at the manual.

Now we all know that the size of print in the TK2 manual is too small to read, so for anyone who would like a more readable copy, I have TK2 on disc, and if you send me a FORMATTED disc, and a Dollar to cover post and packing, I will put this on the disc for you and you can make your own hard copy in whatever size of print you prefer. This is still cheaper than a photo copy enlargement. §

323 1/2 N. Church Street  
Bowling Green, OH 43402  
November 24, 1991

Dear Bob,

I had to write about your "discan" program in the latest newsletter. Naturally it would be faster in ml, but that's another story.

There are two major difficulties I see - maybe more. The first is that you can't use it to find any K mode tokens - NEW through COPY. With a standard INPUT (even compiled) will not ever get into K mode to enter those particular tokens. You could use CHR\$'s if you didn't use the LINE in the INPUT (line 160).

Perhaps you weren't aware of that. If you don't use LINE, then INPUT treats the keyboard entry as an expression and evaluates it. Hence, you could enter CHR\$ 245+"#4" to get PRINT #4. (You would have to backspace or DELETE the quotes that appear on the screen to enter this, of course.)

The other problem is that the search string might appear in a program and your routine not find it because it is split between tracks. This is more likely with a longer string, of course. If the first half of the string is at the end of one track, and the second half at the beginning of the next (in terms of the program's tracks used), your routine can't possibly find it.

The only way around this, even with a revision of the search routine, is to load the tracks of any one program consecutively. You know that they may not be consecutive on the disk, unless no file has ever been erased from the disk.

One other "error", now that I think of it. If you decide to look for something with a number in it, unless the number is the last thing in the string, you will have to deal with the 6-byte "slugs" (the FP part). If all your programs are SAVED after using FP-to-VAL (or something equivalent), you could avoid this, but that's unlikely. I suppose the same applies to embedded colour commands, if they occur in the middle of string in the file and not the search string.

If I were writing this in ml or BASIC, there are two other conveniences I might include. First is to print out the line number in the program instead of the track number. Second, and this would take more work, would be to include a "wild card" character of some kind. This would require there to be some character you know would not be used, which is one of the difficulties.

Of course, if it prints out line numbers, it would not search the variables (or would need a separate routine for this). The code I used in FP-to-VAL could easily have identified the line numbers, if I had thought there was any reason to include it then. So I could easily do that part of it.

A string compare is more difficult in ml. Let me sketch the program out:

Get the search string

```

Load the directory track
  (perhaps store the directory somewhere)
Find the first BASIC program
  Load the first track
  Compute end of track buffer
  Get line number and length.
  If line length+address in memory>end of buffer,
    then empty checked part of buffer, and load next track
  Search to end of line-length of string for match
  If no match, check next line (until variables reached or end
    of file)
  If match, print out program name and line number
Find next BASIC program, and repeat
  (if we didn't store the directory, this means re-LOADing it)
Print "Search completed.", etc.

```

Note the subtraction of the length of the search string - this is to save time. If we are only allowing 1-line searches (which makes sense), this avoids searches which can't possibly match anyway. (Likewise, in your program as written, it is pointless to search past end-LEN s\$+1, since this would put the end of b\$ past the end of the track. See your line 350.)

Oh, while I am back on your program, why do you even need b\$? Replace lines 360 to 380 with:

```

360 IF PEEK n = CODE s$(1) THEN FOR i=2 TO LEN s$: IF PEEK
(n+i-1)=CODE s$(i) THEN NEXT i: GO TO 400

```

This would remove all necessity for b\$, and save you the trouble of setting b\$ to the right length. Provided TimeMachine will accept that line, of course. What do you use blx for anyway? (line 440) The LET nt=x in line 490 is redundant. Etc.

Oh, one other reason for making the program recognize lines. You might be searching for a short string that just happens to match a line number code in the program. Likewise, a string of less than 5 bytes might match a number slug. So the program might say there is a match when there is none unless it is smart enough to recognize lines and slugs, if the search string is less than about 5 characters!

I guess the next step (the way I usually do things) is to "pseudo-code" the program:

```

Input string:
  set up pointers and clear buffer
  Display message "String to search for?" (with ENTER)
  Set start.pos=current.pos, end.pos=current.pos
Main Input Loop:
  Clear screen from start.pos to end.pos
  Display current contents of buffer w/cursor
  Set end.pos=position after printing buffer
  Get key from keyboard
  Special keys:
    GRAPHICS: set G mode
    EXT: set E mode
    EDIT: clear buffer?/K mode?

```



```

        cursor keys
        colours
        DELETE
        ENTER: done input, go to Get Directory
        Other keys (characters): insert into buffer, go to loop
Get Directory:
    Load directory from disk
    Copy BASIC filenames to storage
    Set up pointers
    Copy loop:
        If ERASEd, skip filename
        Check for .B extension
        If found, copy to next storage area
        If not done, loop
    Mark end of directory storage
    Set pointer to first filename in storage
Reset File Buffer pointers (and copy name)
    buffer.end=buffer.start, current.line=buffer.start
Load track
    Erase junk in buffer (if any)
    Copy current.line to buffer.end down to buffer.start
Reset pointers:
    Buffer.end=buffer.end-current.line+buffer.start
    Current.line=buffer.start
Load next track in directory entry (go to Next File if none)
Copy text of track (ignore header) to buffer
Set buffer.end to buffer.end+track.length
Get Line:
    Get line.number from current.line
    Get line.length from current.line
    Compute line.end=address+line.length
    If line.end>buffer.end, go to Load Track
Search1 loop:
    if address+string.len>line.end, go to Next Line
    if char(address)<>string(first), go to loop
    Store start.address
Search2 loop:
    if char(next.address)<>string(next), go to Not Found
    if next<>last, increment pointers and loop
    otherwise, go to Found
    Not Found, restore address=start.address
    increment pointers and loop
Next Line: set current.line=next.line, go to Get Line
Next Track: increment directory pointer, goto Get Track
Found: print filename and current.line
Next File:
    advance file.pointer to next file
    if not directory end, go to Reset File Pointers
Print "Search Completed"
    *(if search again, go to Input(:Display Messages?))

```

That last option (going to Display Messages) and EDIT as clear string would make it convenient to have the previous search as the default. However, if EDIT is used as clear, then we would need something else (cursor up or cursor down?) as K

mode key, perhaps the other as BREAK.

Most of the input subroutine is based on the ROM, but I might suggest a few changes to speed things up. The ROM routine can be slow because it always reprints the entire buffer. In practice, you may only need to print from the cursor to the end or from the start to the cursor (except when you delete a character). This would speed things up if you don't back up. But that would require also knowing the cursor position on the screen.

Note that if you follow something like the subroutine given, you could write a subroutine to allow INPUT #2 to work (that is, INPUT on the upper half of the screen. (I was always surprised they didn't make INPUT #2 work - the only change necessary would be reading the correct position(s).)

This will take a little more time to write than most of my previous stuff. Speaking of which, I did the things I mentioned in my last letter to the clock - displaying initial values and setting attributes. Not that there is anything wrong with the old one, just this one is better. I didn't have time since the last letter to finish MSpack yet. But 3 pages plus the clock is long enough for now.

Take care. Write you again soon.

Sincerely,

*Steven V. Gump*

100 REM Revised clock version 2

102 REM RUN 3000 to enable  
103 REM RUN 4000 to disable  
105 REM to adjust timing,  
change second 60 in  
line 1006 to counted  
seconds per real minute

1000 DATA 33,0,253,17,1,253,1,0,  
1,54,254,237,176,62,253,237

1001 DATA 71,237,94,33,82,255,17  
,31,64,205,201,254,62,10,205,217

1002 DATA 254,205,201,254,62,10,  
205,217,254,126,43,14,0,12,214,1  
0

1003 DATA 48,251,13,198,10,205,2  
17,254,121,213,229,135,135,135,3  
3,128

1004 DATA 61,133,111,6,8,126,18,  
20,35,16,250,122,203,63,203,63

1005 DATA 203,63,246,88,87,21,62  
,7,18,225,209,27,201,0,245,229

1006 DATA 197,213,33,83,255,126,  
60,119,254,60,56,32,175,119,43,1  
26

1007 DATA 60,254,60,56,19,175,11  
9,43,126,60,254,60,56,10,175,119

1008 DATA 43,126,60,254,24,56,2,  
175,119,205,179,254,209,193,225,  
241

1009 DATA 195,56,0,237,86,62,63,  
237,71,201

3000 RESTORE 1000: FOR i=0 TO 15  
3: READ a: POKE 65184+i,a: NEXT  
i

3002 INPUT "INK? ";i,"PAPER? ";p  
: POKE 65271,8\*p+i

3005 INPUT "Row? ";r,"Column?";c  
: LET r1=8\*INT (r/8)

3010 POKE 65207,32\*(r-r1)+c+7: P  
OKE 65208,64+r1

3015 INPUT "time? (hhmmss) "; LI  
NE t\$: POKE 65363,0: POKE 65360,  
VAL t\$(1 TO 2)

3020 POKE 65361,VAL t\$(3 TO 4)

3025 POKE 65362,VAL t\$(5 TO 6)

3030 RANDOMIZE USR 65184: STOP

4000 RANDOMIZE USR 65331: STOP

9900 RANDOMIZE USR 100: SAVE "cl  
ock.Cx"CODE 65184,154

9999 RANDOMIZE USR 100: SAVE "cl  
ock1.BD" LINE 3000

QL

Recently I was asked why at the last minute I had put a D-DUMP procedure into the TorQLib program, when there was a perfectly good SDP\_KEY command that would do the same thing.

The reason was to accomodate those QL'ers who do not have an expanded machine. Yes I know that I said that TorQLib requires an extra 256K. But the library program will work on a straight 128K machine, but only just, And this leaves NO room for you to add your own files to the program. After Archive and TorQLib are loaded, you are left with the region of 7K. Not enough for comfort. If you use TK2, then you are left with just a dot over 6K! (or less according to your configuration)

The SDP\_KEY command does not work without TK2, and there are still some who do not have TK2, thus for anyone wanting to dump one file to the printer, there is not a command available to do it. So I put one in. Simple as that.

#### CHATTER ABOUT 1440 DISC DRIVES---

Just heard from a member that he got some disc drives from Dan Elliott of Cabool at a nice price. \$30 each. I understand they are full height and brand spanking new. I suggest you call first. They won't last long.

#### Chatter about Bill Jones-----

As mentioned earlier Bill has sold most if not all his Sinclair stuff and left the fold. Going in for an IBM 386 system with something like 4 meg and a 140 meg hard drive + + +. Says he is going on a trip to Alaska some time this year. Even talks about the Bering Sea. BBrrringggggg. Sounds like he is going to write a book about it.

I seem to recall him writing some time ago about how much he preferred the warmer climates!

Says to pass on his regards to Louis, and also to Larry Crawford. Message passed.

End of chatter-----Hugh H.

If you like to type in programs slow mode, you will have found that it can be annoying when adding lines to a long program, to have to wait while the computer writes the entire screen every time. It is especially bad if the line is a long Print or Rem line.

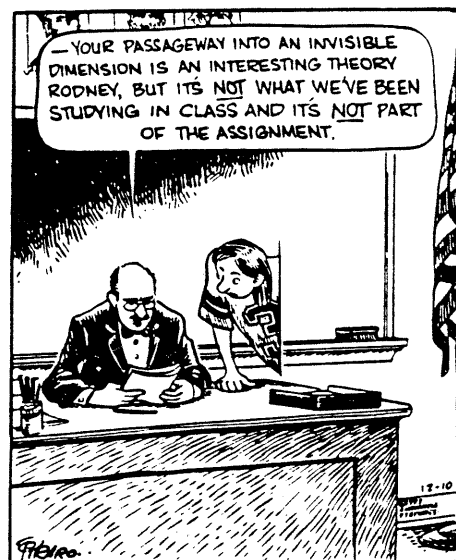
This small program can be temporarily stored near the start of the program using a few spare lines. When one has a screenful of lines, type GOTO (first program line). Answer the prompt with the last line you have entered and it will return you to the last line you have entered and it will return you to that line with an empty screen below. It also POKES the listing system variable to ensure that every relist after that will return to your chosen line.

Retyped from the April '82 issue of Your Computer Vol.2 No.4

```
2 CLS
3 PRINT "TYPE A LINE NO"
4 INPUT XXX
5 POKE 16419, XXX - INT (XXX/
256)*256
6 POKE 16420,INT (XXX/256)
7 LIST XXX
8 STOP
```

#### BIZARRO

By DAN PIRARO



# BYTE POWER

## Super Special!!!

**All BYTE POWER titles at 50% off...**

	<b>.Now.</b>
12 BYTE POWER Magazines (Back issues) .....	\$27.50
6 BYTE POWER Magazines (Back issues) .....	\$16.00
1 BYTE POWER Magazine (Back issue) .....	\$3.00
1 <sup>st</sup> Class Pack (6 back issues in vinyl album) .....	\$20.00
Mega Pack (12 back issues in 2 vinyl albums).....	\$31.50

Issues Available...

August '86, September '86, October '86, November '86, December-January '87,  
February '87, Spring '87, Fall '87, February '88, May '88, Fall '88, Winter '89

BYTE POWER 1 <sup>st</sup> Class Fonts II .....	\$10.00
2+FAST (Fast Load for TS2068).....	\$10.00
The Print Factory .....	\$12.50
The Companion Volume 1 .....	\$12.50
Best of Business and Utilities.....	\$10.00
Best of Arcade Games .....	\$10.00
Best of Entertainment, Strategy & Board Games .....	\$10.00

\*\* For those with remaining issues of BYTE POWER Magazine, you may use them as credit towards the purchase of any of the above packages...

**Please add \$3.00 for Shipping & Handling. U.S. Funds only...**

**BYTE POWER  
1748 Meadowview Avenue  
Pickering, Ontario, Canada  
L1V 3G8**

**SINC-LINK**

Jan/Feb 1992

January 24, 1992

Dear Out-of-Town Members,

I have been busy recently, in fact I'm still at it. writing letters to catch up on my backlog. I'm just about complete, so if anyone is waiting for me, better drop a line.

I am thinking of adding two more disks to the club library. One would be a collection of Spectrum programs which I think are more than a little interesting. Major programs that is.

The second disk would be a collection of articles that have been culled from past issues of our newsletter.

Jeff, our newsletter Editor, is asking for material for the newsletter. Well, he asks the same thing each time a newsletter comes out; complaining that he has used everything he had in this 'latest' newsletter. Well, that's understandable, it has to be made up from material from the members. An interesting thing. The out of town members like the club because of the stimulation they get from the newsletter. While the in-town members get fresh interests from what the OOT members input to us. And keeps us from going stale! A very fruitful association!

If you haven't contributed up to now, please consider this an invitation; no, maybe even a plea!!

I'm considering going into my old correspondence and publishing some of my old treatises that I have written to members answering some of their problems.

Hugh Howie mentioned at our last meeting, that our newsletter should contain more letters from members, stating problems they were having, and asking for solutions. I thought about that for a while. Why has that not happened. Then it struck me. I get questions like that from members, and then directly find the solution and write them back. Without it appearing in the newsletter. Maybe publishing some of these old letters would be interesting. I can think of one or two to D. Bundy, and a few other members that I've written to with material that might be interesting.

Here's some details of newsletters that have come in recently:  
- FDD NEWSLETTER - We've received the Vol. 1 No.5 issue of the FDD newsletter. It contains a lot of information in it's 6 pages that would be of interest to owners of this Disk System. Here is the Editors name, again:

Jay S. Siegel, 1274 49th Street #821, Brooklyn, New York 11219-3011. And his phone number is (718) 853-7521...Sat - Thurs nights, Sunday all day.

Jay says that he hopes to publish monthly, and says that in any case he will put out 12 issues minimum. Depends on how enthusiastic a response he gets from other FDD owners.

- ZXir Clive Alive! - We have also received the Fall 1991 issue of the Timex/Sinclair NorthAmerican User Group's newsletter. Don Lambert has been busy. This issue contains 20 pages of material. Includes a list of current vendors. Also a page of classified adverts, which offer a variety of computers: Timex, Sinclair, CoCo, Amiga, Atari, Commodore 64.

Contributors to the issue are, EMSOFT, Al Feng, Joan Kealy, Don Lambert, Gil Parrish, Bob Swoger. And also John LASTNAMEUNKNOWN who gave a lengthy story about the Dayton ComputerFest.

— The Sinclair Desktop Publishing Journal - I have received a copy of the Vol 2, Issue 4 of the SDPJ, edited by Mike Felerski. This is a 5-page newsletter covering the use of various Desktop Publishing programs. A very good journal for those into DTP. I understand that it is to be placed in the UPDATE magazine. If you would like a copy of it you could also ask me for it.

— The HACKER - Received a copy today. It says in a column heading on the last page.. "HACKER" BITES THE DUST AND BIDS FAREWELL TO ITS READERS . And goes on to say that the club currently has a membership of only 4, and of these only one has a Timex. The circulation of the HACKER was down to six paid members out of 34 mailings. Hence it is ceasing publication.

Not too much of a loss really. They had not published a Timex item for about a year.

For any of our members who have a ZX81/TS1000 with 64K memory and a SRAM card in the 8-16K area, we now have a quite a number of programs ready that we acquired from the Vancouver club. A lot of Fred Nachbaur's and Greg Harder's work in High Res. If you would like a copy on tape, drop me a line and ask for them. Many of them are heavily documented.

Some titles: DUNGEON OF YMIR, ZXPAIN, BLACKJACK, SHREB.BA, F-SCAPES, MANDEL.FF, JULIA-HR, VOYAGE OF PERIL, BIPODS, INVADERS.

I received my income tax forms yesterday, so I have started reworking my tax program. A challenge or a chore, depending on the way you look at it. It is really only suitable for Canada, and then, only for the Province of Ontario. It is for the TS2068. If you are interested in a copy, simply ask.

I've noticed that over the years the demand for 2068 and TS1000 tapes has dropped off almost to zero. And the requests for Larken disks similarly has faded off significantly. Quite a change from those frenzied days when the club first started. Then everyone was typing in programs; now, no-one does. Changing times.

For my part the days of writing Larken utilities has just about come to an end. One feels that they've all been done.

Oh, one other thing. You will notice an advert on the back of our newsletter from BYTEPOWER. I have about three of our members who have not had success in ordering from these people. A matter of orders not being filled, and no response to follow-up letter queries. Robert Shade, Denis Zacharias, and Michael Felerski mention problems. There may be others.

I am following this up with BYTEPOWER, but I have to say, hold off for the moment. I suspect the problem is that they are better programmers than they are administrators. In one case they asked for xerox copies of Money Orders as proof of ordering. Hey, wait a moment, it's not easy to get post office response to cashed(?) money order queries. Have you ever tried it? Watch for further information.

If you've had a problem you could help me by sending complete details.

We are having a flea market at our next meeting. I hope to get rid of some of the junk I've accumulated over time!! It's a bit late to ask if you might be looking for something. But if you are quick enough, I can keep a lookout for things you are interested in.

Sincerely,

George Chambers



REUTER PHOTO

**CALCULATED RISK:** British inventor Clive Sinclair, pioneer of the pocket calculator, holds up his new electric bicycle, designed to take over from pedal power.

## *It's easy, rider, bicycle's electric*

LONDON (Reuter) — A British inventor who pioneered the pocket calculator has produced what he hopes will be another triumph of technology, an electric bicycle.

Sir Clive Sinclair's "Zike" allows tired riders to switch on an electric motor when they don't feel like pedaling any more.

Unveiled at a London trade fair Thursday ahead of mass production in May, it has a lightweight alloy frame with motor and battery. Top motorized speed is 19 km/h (12 m.p.h.).

Sinclair made his name as an early pioneer of

the affordable pocket calculator in the 1970s and his company, Sinclair Research, also helped bring high technology into the home with a personal computer before its fortunes waned in the 1980s.

Sinclair, 51, has since become known for eccentric creations designed to be kind to the environment.

An earlier venture was a futuristic town car, which, like the "Zike," could use either motor or pedal power. The "C5" was steered by handlebars and could reach 24 km/h (15 m.p.h.) using a washing machine engine. But it did not catch on.