SINC-LINK

MAR-APR '91 VOL. 9 NO.2

SINC-LINK VOL.9 NO.2

MAR-APR '91

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.

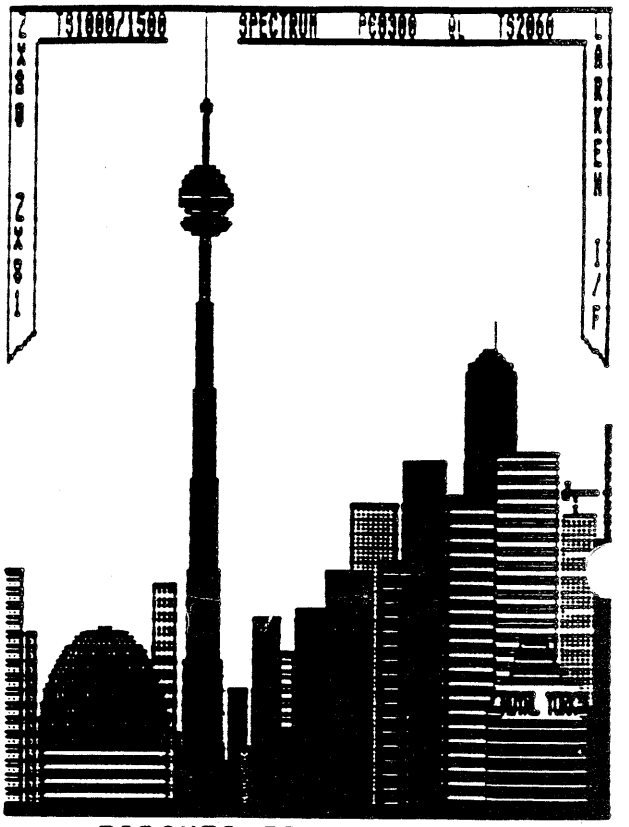
NEWSLETTERS ARE EXCHANGED, FREE OF CHARGE, WITH OTHER TIMEX-SINCLAIR USERS GROUPS.

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. START TIME: 7:00 PM.

SEND CORRESPONDANCE TO:

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



TORONTO TIMEX-SINCLAIR
USERS CLUB

EXECUTIVE OFFICERS:

	(Area Code 416)
PRESIDENT:	RENE BRUNEAU (531-9749)
TREASURER:	BILL LAWSON (444-8772)
SECRETARY:	GEORGE CHAMBERS (751-7559)
ACTIVITIES:	RENE BRUNEAU (531-9749)
TAPE LIBRARIAN ZX81:	LYMAN PAQUETTE (482-4479)
TAPE LIBRARIAN 2068:	RENATO ZANNESE (635-6536)
TAPE/DISK LIBRARIAN QL:	HUGH HOWIE (634-4929)
PAPER LIBRARIAN:	JEFF TAYLOR (244-8583)
NEWSLETTER:	JEFF TAYLOR (244-8583)
LIAISON OFFICER: (Out-of-town members)	GEORGE CHAMBERS, 14 RICHOME COURT, SCARBOROUGH, ONTARIO, M1K 2Y1 (416-751-7559)

TORONTO TIMEX-SINCLAIR USERS CLUB

Editorial

An interesting little get-together was held at Hugh Howie's about three weeks ago. Most of the in-town QL users and the club secretary met with Paul Holmgren for an afternoon bull session and general good time. Paul is best known as a U.S. QUANTA librarian, the current guiding light for SNUG (the Sinclair North American User Group) and a co-owner of Mini Mechanical Affinity, a parts and software company serving the T/S (QL mostly) community.

Everyone happily doled out loot to pick up a variety of bits like keyboard membranes, ROM replacement EPROMS, clock battery-backups and even a TS1000 ROM upgrade.

Then we made Paul really work by grilling him on the progress of SNUG to date and where he thought he would like it to go. Yes, he admitted that things had not moved as quickly as the founding members had hoped and, yes, the newsletter, SNUG Roundup, had been published at rather erratic intervals. Paul attributed this to a combination of the problem of having a club executive spread across the continent, the difficulties of communicating at long distance and perhaps a lack of commitment on the part of some people once the scope of the project was realized.

Anyway all this is about to change, Paul said, because SNUG now has a new newsletter editor who is determined to succeed. To their credit, SNUG has assembled a data base on 2068 disk of sources of information on any topic relating to Timex-Sinclair. Simply ask them to find info on a subject and they will come back with which publications contain the data you require and then they will find the articles for you. It sounds impressive.

The general consensus at the meeting was that after three years of not seeing much for our SNUG membership fees, we will wait and see if SNUG will take off before renewing our membership. I think Paul felt a little defensive after our verbal third degree.

Things then moved on to a question and answer period about making our QLs run better (not crash) and one of the key points Hugh mentioned and Paul agreed with was that the QL is very sensitive to changes in line voltage. Something to keep in mind. Paul was very helpful and forthcoming with answers to our problems and generally everyone came away feeling he had got something useful out the meeting.

Thanks to Hugh and his wife for their hospitality and thanks to Paul for making the trip from Indiana.

Oops

Some of the out-of-town members received the last issue of Sinc-Link with what appeared to be two covers, one after the other and the page numbers in the wrong corners. I assure you that it did not go to the printer like that but since I get it printed for a very reasonable rate I couldn't ask them to redo it. I did however, leave out a listing for Sebastian Boisvert's article. It appears this issue. Sorry, S.B. Oh well, the best laid plans...

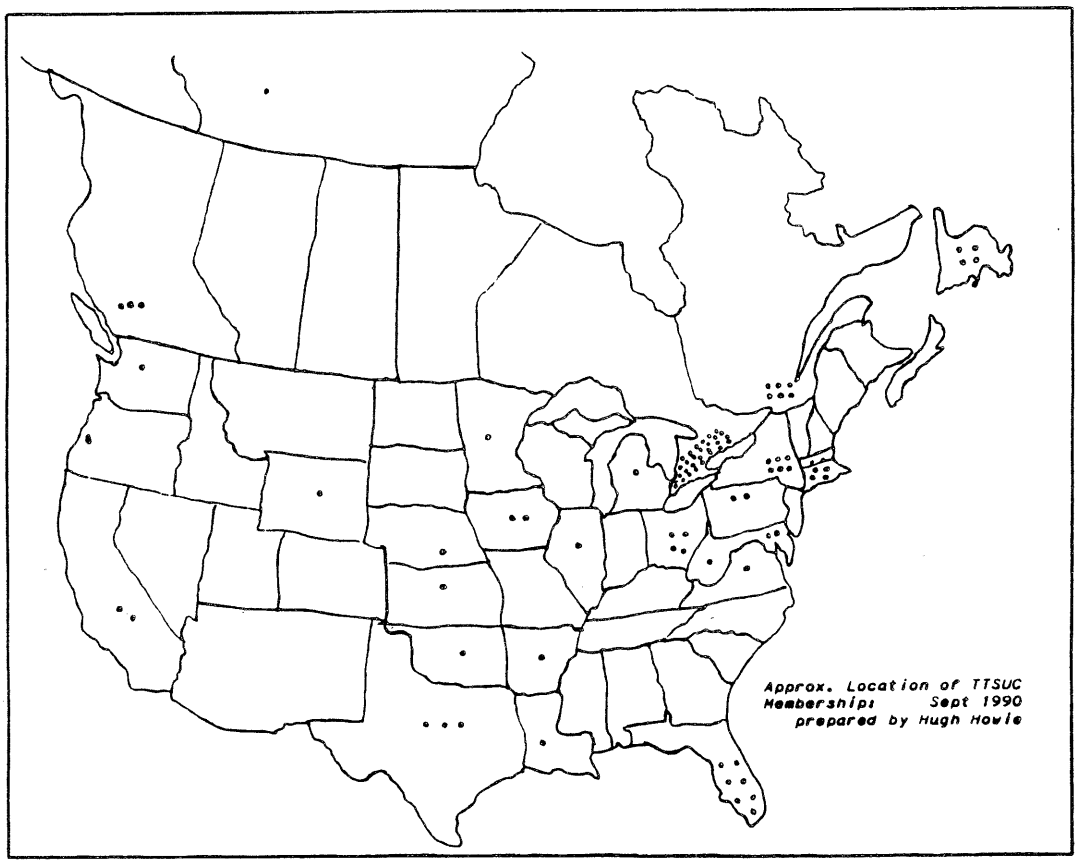
Cont.

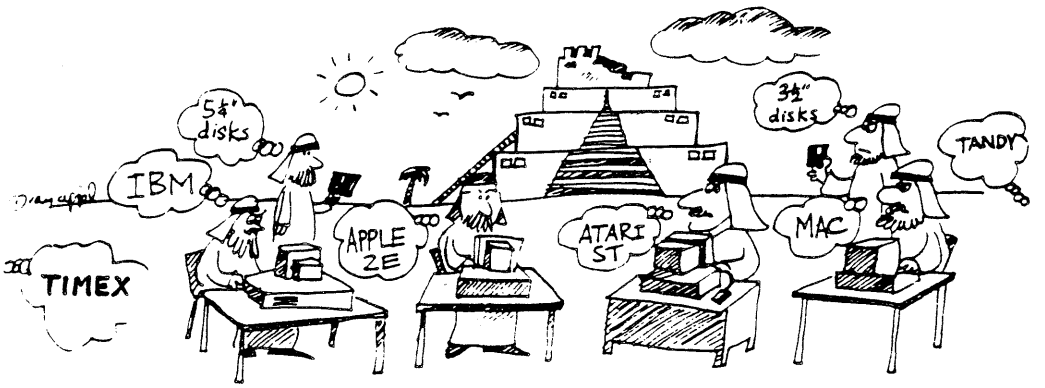
Cover

This issues' cover is another Super Screen from Byte Power's *The Companion* suite of screens for the *The Print Factory* desktop publisher. Not bad for the ol' 2068 and a nine pin printer, eh? Thanks for the positive feedback on last issues' cover, readers.

That's all for now...

J.T.





"And the Lord said, 'Come, let us us go down, and there confuse their language that they may not understand one another's speech . . .'" Gen. 11 V.7.

BOB's NOTEBOOK 910110

In writing this column, I always try to include material from my "notebook" that I have just been working on that should appear in the next Sinc-Link in order to be timely. What follows comes as a result of my latest endeavours.

When I read each issue of Sinc-Link, there is often material that does not interest me at the time ! But as time wears on, I often go back to old issues in search of some information on a subject I was not then into but have now decided to investigate. Thus, I keep a fairly comprehensive index of items for all the issues of the newsletter back to 1985. This I keep in two Pro/File files, slink and slink2 which are available from the club library.

For those using Pro/File, I have just performed some minor surgery on the BASIC to put in a couple of options that may interest you:
(1) There is now a CAT option that displays the directory on the disk in the selected drive; then goes on to display more information in the form of start address and length of each file on that same disk (a.k.a. Fullcat);
(2) There is also an option to allow you to switch back and forth between the Left and Right side of the 8.5 inch page. This seems a rather crude way to pack more material onto each page for after printing the left column for each page, it is necessary to roll the paper back and then print the right-hand column.

Another way I can see to do this would be to export the Pro/File file into Pixel Print Professional and take advantage of the Bank Switching technique to print the left and right columns simultaneously. I tried this approach; if you want to have a go, here are a few lessons learned:
(1) SAVE the Pro/File file to disk using the RND/SEQ commands in the Version 3 EPROM. Give the new files a name ending with the Mscript extensions (.CT) eg, you would open a channel thus:
RANDOMIZE USR 100: OPEN #3,"name.CT OUT".
(2) In PPP, the MSCRIPT to PP conversion routine will only handle up to 375 lines, so you must save text in parts that do not exceed this limit. You can accomplish this by using the Profile search command wisely, eg, using 89 will save all the records with that number in them.
(3) Blank lines will not be saved so don't count them as you keep track.
(4) Load PPP and select the Mscript conversion routine. Use the 64 col option.
Etc. Etc. Etc.

I concluded it is far easier to use the Left/Right toggle and roll the paper back.

In any case, I have provided a copy of the revision for the club library and it is available to those who are valid owners of Pro/File.

One tack I have taken lately (suggested to me by Richard Hurd) was to disconnect my TS2040 and cope only with the wide printer. So far, it has been successful with the only problem area being getting a printout of Timachine output and this can be done using the Random/Sequential Filing commands. You may recall that I covered that in the May/Jun 90 issue page 4.

Let's consider interaction between files:

One of the side-effects of using Omnibus is that sometimes loading a program from the menus produces strange results or no results at all.

For example, if you save Timachine onto the Omnibus disk and try to load it using a menu option, you may be frustrated. It seems that some incompatibility exists that prevents Timachine from being loaded unless the computer is reset or at least another AUTOSTART file is loaded. I have never been able to discover the cause or how to resolve this problem.

However, one problem occurred recently which I was able to resolve. If you have Mscript (tailored for Omnibus/Ramdisk use) as an Omnibus menu option, load it and check how many bytes have been used (<LENGTH> on the Mscript home menu); it should be 1 (one). Now quit Mscript which should take you back to the Omnibus menu. Load some other program, say, Profile; then quit to the Omnibus menu. Now reload Mscript. If you do not get a length 1 on the menu, you have this incompatibility. For the length 1 to be displayed, I have discovered that the addresses 46927, 35440 and 44935 must all hold zeroes. Go back to the Omnibus file and break into BASIC. Find the line that loads Mscript; it should be 3320. After the first PAUSE oo, add

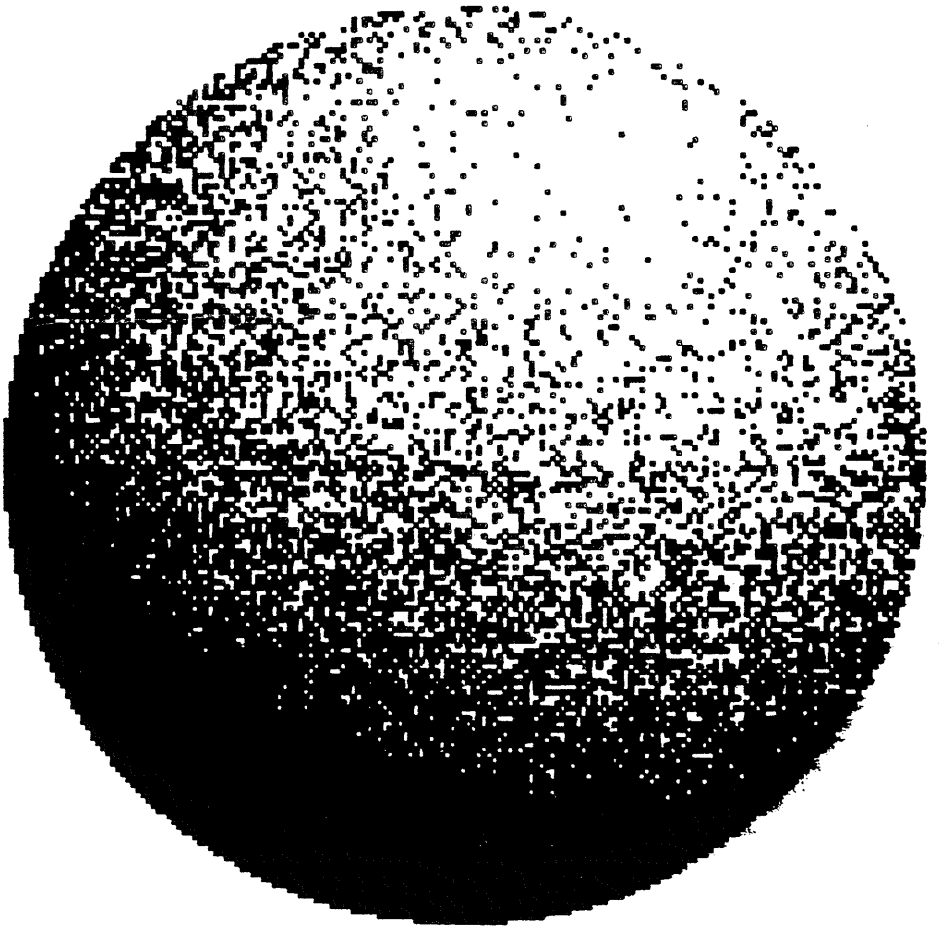
<POKE 46927,oo: POKE 35440,oo: POKE 44935,oo>. Now each time you load Mscript, no matter what you were doing before, you should get the correct Length reading.

If you keep Mscript and Timachine on their separate disks and load them via their own Autostarts, this problem will just not occur. It's the interaction between these programs and Omnibus that causes the clash.

It's the curse of complexity; but that's what make programming the TS2068 challenging and good fun, don't you agree? Let's hear from you readers who have anything to add to all this stuff this time.

Bob Mitchell 20 Wild Briarway Willowdale Ont M2J 2L2 494-5260.

TORONTO
TIMEX-SINCLAIR USERS CLUB



```
  1 REM This program will produce the globe above.
 10 FOR x=-87.5 TO 87.5: LET i=INT (SQR (7744-x*x)-.5)+.5
 20 FOR y=-i TO i: IF x+2*y+2*SQR (7744-x*x-y*y)<RND*264 THEN
PLOT x+127.5,y+87.5
 30 NEXT y: NEXT x
```

QLIPS

By Hugh H Howie

QL BASHERS_____

Why are there so many QL Bashers out there? Some have never even tried the QL. Anyone I have spoken to, and has used the QL has nothing but praise for it. The main gripe is that it is not IBM compatible. Most folks I have spoken are very quick to praise the power of the QL. Except in a very few cases it is lauded as being better than most PCs on the market, even to-day.

There is nothing wrong with the QL other than the fact it was pushed onto the market before it was ready. Even to-day, the QL is a powerful machine. Don't let anyone try to tell you otherwise.

What prompted me to write this, was that last week I received a letter from Marinus Heuseveldt in Florida, the only person who put pen to paper in reply to my suggestion of a QL Hardware Library, and also Tuition Course. I would like to Quote from his letter:-

"Your idea of a hardware library (QL) sounds good to me, but there may be a question of how much use it will get. It is one thing to hardware hack a TS1000- still another the 2068, and yet another, a QL. To the hardware hacker there wouldn't be any reluctance to work the QL over, however, to the less capable, experienced, there would be"

He also pointed out a few difficulties with a correspondence course in programming. A positive letter with many valid observations. Very constructive. Very welcome. I would like to thank Marinus for his well argued comments, and for taking the time and trouble to write. This article was prompted by his letter to me. Just demonstrates what participation in communication can do. We need more of it. Constructive stuff.

In other words, what he is saying is that the QL is a much more complex machine to hack than the 2068.
So there----------!

Here is another quote, this time from a well known programmer of QL and IBM experience. His experience is mainly in the ARCHIVE field, but he is also well versed is all aspects of the QL. Bill Cable is a prolific contributer to UPDATE, and his articles are always looked forward to by the QL fraternity. In other words he is an expert, probably one of the best in the world on the QL. This extract is from the April 1990 issue of UPDATE.

Bill Cable writes:-

"How do we judge the QLs performance against the other machines? 1. QL much slower in its screen handling. This is evident in QUILL and ARCHIVE. I tried SPEEDSCREEN and LIGHTNING but they didn't improve the times. Some programs still manage to give fast screens. The EDITOR and TEXT87 are examples. Also the QL was multitasking QUILL while DBEasy was being run. That didn't affect the speed of doing the tasks because QUILL was sitting idle. None of the other computers can multitask. None of the other computers can network either, without expensive equipment. And the AT machines cost 2 to 5 times as much as the QL. At some point I will try these same tasks on the THOR and pass on the results. It should be much closer to the AT machines. If you were to judge the QL purely on the response times it shows its age. If you judge it on its overall features taking into account its powerful operating system, its cost, and how much you can accomplish with it then it still can hold its own against most PCs."
End of Quote.

That is what the *experts* think.
(me too)

The DBEasy program Bill mentions is one of his own creations, and is a database manager for ARCHIVE, permitting the creation and managing of databases for many applications, allowing switching between databases. He has also written a great many other Archive Programs worthy of your attention. He is a regular contributer to UPDATE Magazine, called "The Cable Column"

May his shadow never grow less.

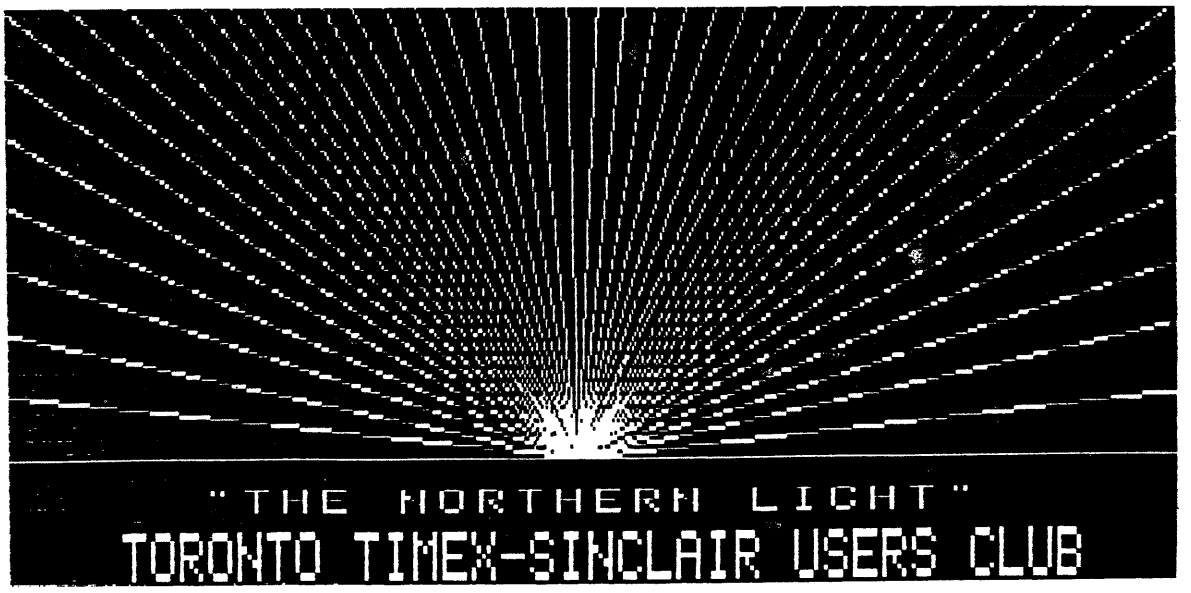
§§§

LISTING FOR DISK MENU MAKER

```
  1 REM DISK MENU 910216 LDC
  2 REM    based on the program    developed by George Chambers
  & Bob Mitchell
  3 PRINT CHR$ 2: CLS : ON ERR  RESET
  4 RANDOMIZE USR 100: OPEN #4,"dd"
  5 LET c=100: BORDER 6: INK 9: PAPER 5: CLS
 10 DIM X$(16): FOR i=0 TO 15: LET q=PEEK (26660+i): IF q=0 THEN
 GO TO 15
 11 LET x$(i+1)=CHR$ q: NEXT i
 15 PRINT AT 1,1; PAPER 1; INK 9;"MENU FOR DISK "; PAPER 2;x$''
 19 RESTORE c: LET n=0: PRINT
 20 READ a$: READ b$: IF a$<>"STOP" THEN  LET n=n+1: GO SUB 7000
: GO TO 20
 35 PLOT  INK 3;0,0: DRAW 255,0: DRAW 0,175: DRAW -255,0: DRAW 0
,-175: PLOT 1,1: DRAW 253,0: DRAW 0,173: DRAW -253,0: DRAW 0,-173
: INK 9
 38 PLOT 9,20: DRAW 237,0: DRAW 0,135: DRAW -237,0: DRAW 0,-135
 50 PRINT AT 20,7; PAPER 2; FLASH 1;"PRESS A TO ";CHR$ (n+64);"
ONLY"; FLASH 0
 70 POKE 23658,8: PAUSE 0: LET n$=INKEY$
 80 IF n$<"A" OR n$>CHR$ ((n+1)+64) THEN  GO TO 70
 85 POKE 23658,0: BORDER 1: PAPER 5: INK 9: CLS
 90 RESTORE 100: FOR i=1 TO CODE n$-64: READ a$: READ b$: NEXT i

 91 LET t$=b$(LEN b$-1 TO )
 92 IF t$="CT" THEN  GO SUB 5E3: RANDOMIZE USR c: PRINT b$: PRIN
T CHR$ 2: PRINT "END OF FILE...PRESS ANY KEY": PAUSE 0: GO TO 1
 94 IF t$(1)="B" THEN  RANDOMIZE USR c: LOAD b$
 98 RANDOMIZE USR c: LOAD b$CODE : STOP
1001 DATA "STOP"," "
1002 GO TO 1
3200 RANDOMIZE USR c: GO TO 4: PRINT #4: NEW
5000 INK 0: PAPER 7: BORDER 7: CLS
5010 RANDOMIZE USR c: LOAD "taswi.Cx"CODE
5020 RANDOMIZE USR 64300: PRINT CHR$ 3;
5030 ON ERR  GO TO 1: RETURN
7000 IF LEN a$<24 THEN  PRINT "  ";CHR$ (n+64);" - ";a$: RETURN
7010 PRINT "  ";CHR$ (n+64);" - ";a$( TO 24): PRINT TAB 6;a$(25 T
O ): RETURN
8000 CLEAR 65535: RESTORE 8010: FOR i=0 TO 32: READ a: POKE 24311
+i,a: NEXT I: RANDOMIZE USR 24311: RUN
8010 DATA 243,205,98,0,175,50,29,32,50,32,32,205,126,0,205,123,0,
33,4,50,17,36,104,1,16,0,237,176,58,100,0,251,201
8887 STOP
8888 CLS : INPUT AT 10,0;"DESTINATION DRIVE #? ";dd: LET c=100: R
ANDOMIZE USR c: POKE 8200,8195: LET s=USR 110: LET sd=INT (s/4-(s
>8)): RANDOMIZE USR c: GO TO sd: RANDOMIZE USR c: LOAD "taswi.Cx"
CODE : RANDOMIZE USR c: GO TO dd: RANDOMIZE USR c: SAVE "taswi.Cx
"CODE 63223,1492
8889 DELETE 8888,8889
9999 CLEAR : POKE 23730,PEEK 23641: POKE 23731,PEEK 23642+1: CLS
: PRINT AT 10,0;"PRESS <D> KEY FOR AUTOSTART SAVE": RANDOMIZE USR
 102: GO TO 8000
```

Periodically its nice to set up your ZX81 and have a little fun.
I like to think of games in three categories. 1) ARCADE - these are the kill the cleons type, pong etc. In the early days of home computing these games flourished but today there is no way to compete with Nintendo. 2) ADVENTURE - these are the games were you try to find your way interactively through the castle and get the treasure. I have found a few good examples that run on my ZX81, for example TREASURE HUNT. 3) MIND - these are the interesting games because they do not require a great deal of graphics just a good thinking mind, for example THE TOWERS OF HANOI. It is the last two types of games that I am searching for. I have looked through our club library and can find several examples of the ARCADE games and a few ADVENTURE games, but very little MIND games. I suspect that there are a lot of these games gathering dust in the bottom of your filing cabinets but I would like to see them. Have you got some great MIND game that you think would stump the whiz? Heres the deal. Send me whatever you have in the way of MIND or a really good ADVENTURE game, either listing or on tape. I will edit and accumulate them into a set of games. If you send me something, you will get a copy of the complete set of all games sent in.
Send your stuff to:
Ron Campbell
37 Sir Galahad Place
Markham, Ontario
L3P 3K7



323 1/2 N. Church Street
Bowling Green, OH 43402
November 19, 1990

Dear George,

I got the November Sinc-Link today, and thought I would write you concerning packing numbers, as was discussed in the TS Bulletin, p. 4. (By the way, there was no page 2 on the copy I got. Others may have had that problem as well.)

In general, you can place 2 numbers into 1 byte, but not 3 as was attempted by this routine - unless you can't use every number. However, you can fit 7 digits in 3 bytes, by extending the normal method used to double-POKE. This would look as follows: (The variable ph is presumed to be the phone number.)

```
  1 REM (at least 15 characters)
  5 REM Telephone number storage
 10 FOR i=1 TO 3
 15 LET hi = INT (ph/256)
 20 POKE 16524+i,ph-hi*256
 25 LET ph = hi
 30 NEXT i
 40 REM handles any 7-digit number
 50 PRINT "TEL NO. ";
 60 FOR i=3 TO 1 STEP -1
 70 LET ph=256*ph+PEEK(16524+i)
 80 NEXT i
 90 PRINT ph
 99 STOP
```

I should note that this routine would handle any kind of a number provided it was less than 16,000,000. As for marking invalid (non-existent) numbers, there are over 6 million left over, so that should be easy.

Actually, there are fewer phone numbers than the 10 million I was presuming above. First, no 7-digit phone number can begin with a 0 or 1, the operatop or long-distance codes. Secondly, the first 3 digits must not correspond to an area code, or the phone company's switching circuits would get confused. Since all area codes have a second digit of 0 or 1, we could compute there are really only 6,400,000 valid 7-digit phone numbers. It still takes 3 bytes to store it, but we have a lot of spare digits.

If we wanted to use 10-digit phone numbers, we could use this to our advantage. Normally, 10 digits would require 5 bytes. But there are enough un-usable numbers so that we could get by with only 4, and have the entire area code in 1 byte. This requires a little more trickery, but not much. In the following part of a program, presume (ac) is the area code, and (ph) is the phone number:

```
  1 REM (at least 15 characters)
  5 REM Store (ac) and (ph)
 10 LET ac=ac-80*INT (ac/100)
 15 POKE 16524, ac
```

```
20 FOR i=1 TO 3
25 LET hi=INT(ph/256)
30 POKE 16524+i,ph
35 LET ph=hi
40 NEXT i
45 REM phone number stored
50 PRINT "TEL NO. ";
55 LET ac=PEEK 16524
56 LET ac=ac+80*INT (ac/20)
60 IF AC>0 THEN PRINT "(";ac;")";
65 FOR i=3 TO 1 STEP -1
70 LET ph=ph*256+PEEK(16524+i)
75 NEXT i
80 IF ph>0 THEN PRINT ph
85 IF ph=0 THEN PRINT "None"
99 STOP
```

Of course, you would need a line to put the phone number into ac and ph. I am using a 0 in either to indicate it is missing, since neither can be all 0's.

Similar techniques could be used in any field with data from a restricted set of characters - anything numeric, or even postal codes. If we wished to use end-of-field markers and such, we could change every 256 to a lower number, like 250, or anything down to about 220. Anything below 220, though, and our 7-digit phone number would need at least 4 bytes of storage, and 1 for the area code.

Well, that's all for right now. Also note that on a 2068, we would change the 16524 to some other number. In a practical program, you would want a variable instead - we can't POKE everything into the same location. Or we could change the POKEs and PEEKs to use CODE and CHR$ and put the number into a string or string array.

I am almost done with my version of Statistical Analysis for single variable distributions, which I am writing for Bob Mitchell. The program he had, which he said he pulled out of a magazine, was a mixture of different routines for one and two variables, which made it confusing to use, so I thought I would seperate them and clear it up some.

Anyway, I will probably write you again in a couple of weeks when I am satisfied with it. I'll be in Windsor this week, but I'll be back by next week - you may remember I always go to Windsor for my American Thanksgiving break. So bye for now, and peace.

Sincerely,
Steven V. Gutknecht

THE QUEBEC LINK

by: Réal Gagnon
8286 St-Hubert
Montreal (Que)
Canada H2P 1Z3

The last version of ARCHIVE, called ARCdev and ARCrtm, introduces the possibility to call external machine code routine to complement the ARCHIVE programming language. This is also true for the PSION XCHANGE software.

We load the external routines, with LOAD USR 'filename'
the default extension is '_pmc'.

and we call them with LET x=USR(par1,'par2$').

The first parameter, PAR1, is a number and the second one (PAR2$), a string. Note that even if PAR2$ is a variable, we have to put quotes around it. The variable X will contain an error code in the QDOS tradition.

From the programming point of view, the routine must begin with a 6 bytes header like this: DC.B 'pmc0'
DC.W routine_length

The register D0 contains the value of PAR1 and A0 is pointing to PAR2$. The first byte of PAR2$ is length of the string.

Only one '_pmc' file can be in memory. But it is possible to develop a toolkit for ARCHIVE and use PAR1 to select the different routines by testing the value of D0 in our program:

```
ex.   ROUTINE_0   DBF d0,ROUTINE_1
                    ...
                  RTS
      ROUTINE_1   DBF d0,ROUTINE_2
                    ...
                  RTS
      ROUTINE_2   DBF d0,ERROR
                    ...
      ERROR       MOVEQ #-15,d0
                  RTS
```

PAR2$ can pass more than one parameter by using the ascii code of each character. With this method each parameter can have a value between 0-255. For example, if the routine 2 of our imaginary toolkit is to produce a beep, the first parameter can be the tone and the second, the length. For example, if we need a tone of 20 of length 5, this can be coded with LET beep$=CHR(20)+CHR(5)
and called with LET x=USR(2,'beep$').

The following utility will permit to ARCHIVE to communicate via the SER2 port. We will use it to dial a phone number by sending the appropriate command to a modem.

In ARCHIVE, load SER2_pmc with LOAD USR 'SER2'. If you want to dial a number and your modem is HAYES compatible then type
LET number$='ATDT1234567'+CHR(13) .

'ATDT' is a command to dial a number on a TOUCH TONE line, if you have a PULSE line use 'ATDP' instead. You have to terminate the variable number$ with CHR(13) to force to modem to execute the command.

To start the dialing type LET x=USR(speed,'number$') where speed represents to baud rate specific to your modem. Speed equals to 0 for 75 bauds, 1 for 300, 2 for 600, 3 for 1200, 4 for 2400, 5 for 4800, 6 for 9600 and 7 for 19200!

While the modem dials the number pick up the phone. After the dialing, the modem must hang up the phone. Just send a carriage return to the modem, this will put it off line right away.

```
let number$='ATDT1234567'+chr(13)
let hangup$=chr(13)
let x=usr(3,'number$')
print 'Press a key to put the modem
      off line after the dialing.'
input dummy$
let x=usr(3,'hangup$')
```

Say you have a database containing a membership list for your bowling club. The phone number of each member is given by the field phone$. To call Joseph, find his record and issue the following commands to dial his number:

```
let number$='ATDT'+phone$+chr(13)
let x=usr(3,'number$')
```

To create SER2_pmc, type in the BASIC loader and run it.

I repeat, this will not work with ARCHIVE 2.3 or 2.35. You need ARCdev/ARCrtm (also called ARCHIVE run-time) or the PSION XCHANGE which is a combinaison of QUILL/ARCHIVE/EASEL/ABACUS into one big program. I have developped SER2_pmc with XCHANGE v3.27, I don't have ARCdev...

SER2_pmc BASIC LOADER

```
100 REMark LOADER for SER2_pmc
110 REMark  by Real Gagnon, Montreal Quebec dec90
120 :
130 CLS:INPUT "SAVE SER2_pmc WHERE? (default flp1_)?";sauve$
140 IF LEN(sauve$)=0:sauve$="flp1_"
150 PRINT "One moment please.";
160 :
170 RESTORE 1000
180 memoire=RESPR(512):base_memoire=memoire
190 compte=0 : checksum=6567 : checkme=0
200 :
210 REPeat loop
220   IF EOF:EXIT loop
230   READ byte:POKE memoire,byte:memoire=memoire+1
240   compte=compte+1
245   checkme=checkme+byte
250   PRINT ".";
260 END REPeat loop
265 :
270 IF checkme<>checksum:PRINT \"Checksum incorrect":STOP
280 :
290 SBYTES sauve$&'SER2_pmc',base_memoire,compte
300 PRINT 'Ok!':STOP
310 :
1000 DATA 112, 109,  99,  48,   0,  86,  38,  72,  65, 250
1001 DATA   0,  66,  50,  24,  81, 200, 255, 252, 112,  18
1002 DATA  78,  65, 114, 255,  38,  60,   0,   0,   0,   0
1003 DATA  65, 250,   0,  36, 112,   1,  78,  66,  74,  64
1004 DATA 102,   0,   0,  24,  20,  27,  72, 130, 118, 255
1005 DATA  34,  75, 112,   7,  78,  67,  74,  64, 102,   0
1006 DATA   0,   6, 112,   2,  78,  66,  78, 117,   0,   6
1007 DATA  83,  69,  82,  50, 105, 114,   0,  75,   1,  44
1008 DATA   2,  88,   4, 176,   9,  96,  18, 192,  37, 128
1009 DATA  75,   0
```

The Quebec Link_

```
*********************************************************
* filename : SER2_pac  version 1.0                      *
*     date : december 1990                              *
*                                                       *
* Machine code extension for ARCHIVE 2.38 and XCHANGE   *
* to add the possibility to talk directly to the        *
* SER2 port.                                            *
*                                                       *
* LOAD with the command   LOAD USR 'ser2'               *
* USE  with the command                                 *
*       LET result=USR(speed,'string$')                 *
*        where speed is the baud rate to be used        *
*          0=75   1=300   2=600   3=1200   4=2400       *
*          5=4800 6=9600  7=19200                       *
*        where string$ is the variable name             *
*          containing the string to be sent.            *
*          note that you have to put quotes around      *
*          it.                                          *
*                                                       *
*        the routine returns an error code in the       *
*        variable result if any.     0 = no error       *
*                                  other = error        *
*                                                       *
* This program is donated to the public domain          *
*                                                       *
* written by Real Gagnon      and   HIC LOGIQUE         *
*            8286 St-Hubert         P.O. Box 1393       *
*            Montreal (Quebec)      Place du Parc       *
*            CANADA  H2P 1Z3        Montreal Quebec     *
*                                   CANADA H2W 2R4      *
*********************************************************
```

```
MT_BAUD         equ     $12             set the baud rate
IO_OPEN         equ     $1              open a channel
IO_CLOSE        equ     $2              close a channel
IO_SSTRG        equ     $7              send a string to a channel
THIS_JOB        equ     -1

header          dc.b    'pac0'
                dc.w    finish-start
*---------------------------------------
*               A0 contains the pointer of the string to be sent.
*               D0 contains the baud-rate to be used
*---------------------------------------
*
*---------------------------------------
*               set the baud rate
*---------------------------------------
start           move.l  a0,a3           preserve the string address
                lea     baud_choice,a0  find the correct baud rate
baud_loop       move.w  (a0)+,d1
                dbra    d0,baud_loop
                moveq   #MT_BAUD,d0
                trap    #1
*---------------------------------------
*               open a channel for SER2
*---------------------------------------
                moveq   #THIS_JOB,d1
                move.l  #0,d3           we open a channel for a device
                lea     device_name,a0
                moveq   #IO_OPEN,d0
                trap    #2              open the channel
                tst     d0              error?
                bne     error_detected
*---------------------------------------
*               A0 contains the CHANNEL ID  (needed for IO_SSTRG)
*               now we have to send the string pointed by A4
*---------------------------------------
                move.b  (a3)+,d2        length of the string
                ext.w   d2
                moveq   #-1,d3          no timeout
                move.l  a3,a1           string address
                moveq   #IO_SSTRG,d0
                trap    #3
                tst     d0
                bne     error_detected
*---------------------------------------
*               now, we close the channel (A0 still contains CHANNEL ID
)
*---------------------------------------
exit_send       moveq   #IO_CLOSE,d0
                trap    #2
error_detected  rts                     go back to ARCHIVE/XCHANGE

device_name     dc.w    6               length of the device name
                dc.b    'SER2ir'        no handshake and no parity

baud_choice     dc.w    75,300,600,1200,2400,4800,9600,19200

finish          end
```

MIKE'S NOTEBOOK

By: Michael J. Di Rienzo

If you are used to SAVEing SCREEN$ files, you may have noticed that the file size is rather large especially if you LOAD or SAVE using cassette storage medium. The file is 6912 bytes long. Another problem to overcome is that if you want to store more than four or five SCREEN$'s in RAM memory, there is hardly any memory left for your programs. This months utility will COMPRESS a typical SCREEN$ file up to as much as 1/2 its usual size. In addition, a header is automatically included in the SAVED file which will resolve the compressed SCREEN$ back to its full size onto the screen by simply using RANDOMIZE USR (address). The following program will SAVE the compressed SCREEN$ CODE at address 40246 and you will be shown the file length if you wish to write it down. When you are ready to LOAD the compressed file into your own program, you can LOAD it to any legal address then "call" that address to get your expanded SCREEN$. The following program will POKE in the CODE then prompt you to INPUT the file name of the SCREEN$ that you want to compress. It will then LOAD it and temporarily store it. Next you'll be asked what name you want to give the compressed SCREEN$. The SCREEN$ will be instantly compressed and you will be notified of the start address and byte length of the file. Now, a prompt to press any key when you are ready to SAVE your file. Don't worry, your compressed file is already stored at address 40246. This utility can be easily modified by you to work with any disk system. I use this utility alot to animate as many as 15 compressed SCREEN$ by LOADing them consecutively in memory then calling them in sequence to simulate animation. The CODE is not relocatable and is 132 bytes long. Have fun!

"CMPRES"

By Michael J. Di Rienzo

```
  10 CLEAR 36611: LET store=3672
0: LET recall=36732: LET compres
s=36612: GO SUB 100
  20 INPUT "What is the file nam
e to LOAD?  ";n$
  30 LOAD n$SCREEN$ : RANDOMIZE
USR store
  40 INPUT "What is the file nam
e to SAVE?  ";n$: RANDOMIZE USR
recall
  50 RANDOMIZE USR compress: LET
 L=USR 40192: LET L=L-40246
  60 PRINT INK 2; PAPER 5;AT 10,
0;"File:""";n$;"""CODE 40246,";L
; FLASH 1;AT 11,5;"Press any key
 to SAVE": PAUSE 0
  70 SAVE n$CODE 40246,L
  80 CLS : RANDOMIZE USR 40246:
PRINT INK 1; FLASH 1;AT 10,0;"To
 resolve the compressed file ba
ck to the screen in the futureRA
NDOMIZE USR (Starting address wh
ere CODE is LOADed!)"
  90 STOP
 100 LET t=0
 110 FOR N=36612 TO 36743
 120 READ a: POKE n,a: LET t=t+a
 130 NEXT n: IF t<>12070 THEN PR
INT FLASH 1;"Data Error! Recheck
 DATA line.": STOP
 140 RETURN
 150 DATA 17,0,157,33,16,143,1,9
8,0,237,176,201,197,209,33,100,0
,25,235,33,0,64,14,1,126,167,40,
9,254,255,40,5,237,160,3,24,21,7
1,121,60,35,40,7,126,184,32,3,12
,24,243,120,18,19,121,18,19,14,1
,124,254,91,32,217,213,193,201,1
7,0,64,221,33,46,0,221,9,122,254
,91,200,221,126,0,167,40,10,254,
255,40,6,18,19,221,35,24,236,221
,70,1,18,19,16,252,221,35,221,35
,24,223
 160 DATA 17,0,157,33,0,64,1,0,2
4,237,176,201,17,0,64,33,0,157,1
,0,24,237,176,201
```

```
compress 36612 8F04 LD DE,40192
         36615 8F07 LD HL,36624
         36618 8F0A LD BC,98
         36621 8F0D LDIR
         36623 8F0F RET

         36624 8F10 PUSH BC
         36625 8F11 POP DE
         36626 8F12 LD HL,100
         36629 8F15 ADD HL,DE
         36630 8F16 EX DE,HL
         36631 8F17 LD HL,16384
         36634 8F1A LD C,1;
         36636 8F1C LD A,(HL)
         36637 8F1D AND A
         36638 8F1E JR Z,36649
         36640 8F20 CP 255;
         36642 8F22 JR Z,36649
         36644 8F24 LDI
         36646 8F26 INC BC
         36647 8F27 JR 36670

         36649 8F29 LD B,A
         36650 8F2A LD A,C
         36651 8F2B INC A
         36652 8F2C INC HL
         36653 8F2D JR Z,36662
         36655 8F2F LD A,(HL)
         36656 8F30 CP B
         36657 8F31 JR NZ,36662
         36659 8F33 INC C
         36660 8F34 JR 36649

         36662 8F36 LD A,B
         36663 8F37 LD (DE),A
         36664 8F38 INC DE
         36665 8F39 LD A,C
         36666 8F3A LD (DE),A
         36667 8F3B INC DE
         36668 8F3C LD C,1;
         36670 8F3E LD A,H
         36671 8F3F CP 91; [
         36673 8F41 JR NZ,36636
         36675 8F43 PUSH DE
         36676 8F44 POP BC
         36677 8F45 RET

         36678 8F46 LD DE,16384
         36681 8F49 LD IX,46
         36685 8F4D ADD IX,BC
         36687 8F4F LD A,D
         36688 8F50 CP 91; [
         36690 8F52 RET Z
         36691 8F53 LD A,(IX+0)
         36694 8F56 AND A
         36695 8F57 JR Z,36707
         36697 8F59 CP 255;
         36699 8F5B JR Z,36707
         36701 8F5D LD (DE),A
         36702 8F5E INC DE
         36703 8F5F INC IX
         36705 8F61 JR 36687

         36707 8F63 LD B,(IX+1)
         36710 8F66 LD (DE),A
         36711 8F67 INC DE
         36712 8F68 DJNZ 36710
         36714 8F6A INC IX
         36716 8F6C INC IX
         36718 8F6E JR 36687

store    36720 8F70 LD DE,40192
         36723 8F73 LD HL,16384
         36726 8F76 LD BC,6144
         36729 8F79 LDIR
         36731 8F7B RET

recall   36732 8F7C LD DE,16384
         36735 8F7F LD HL,40192
         36738 8F82 LD BC,6144
         36741 8F85 LDIR
         36743 8F87 RET

         36744 8F88 LD DE,13
```

CMPRES.B1 m/c

QL MEANDERINGS and ONE-LINERS

by Hugh Howie.

'Tother night I was sittin thinkin. Usually I just sit. 'tis easier.

A long time ago I asked you to send in some ONE-LINERS, nothing came of the request, but I still think there are a lot of one-liners out there just waiting for who-ever to send them in. So git going guys (and dolls also of course)

While sittin thinkin, I came up with some thoughts might be of interest to someone. Don't know who, but you never know in this Hi-Teck world we live in just who might be in a mood to listen to some-one who is sittin and thinkin rather than some-one who is just sittin. If you is sittin thinkin you might just say somethin worth hearin. That is if you has a mind to talkin. Here is some of my thinkin, if you has a mind to listnin.

Say you write a lot of letters and save them to Disk or MDV, giving them names according to whom you are writing. (Almost said "you are writing to!" (never end a sentence with a preposition.) All the time I do it.

(Propositions are different!)

When you want to recall when and what you wrote, you have to go through a whole string of names on the DIR of your Disk or MDV. Are you still with me?

But TK2, including Trumpcard, has a command to help you. Say you write to Tom or Dick or Jack. (ha ha you thought i was going to say Harry) Anyway you want to recall the letters to Dave. All you have to do is type in:-

"DIR FLP1_Dave"

That is if you use "Dave" as the reference. After the 'ENTER' you wait a second and what do you know, up on the screen comes a list of the letters you wrote to Dave. Try it for the experience.

Here is another little gem. You are sending something to the printer, graphics or something like that, and you want to slip in a little comment at a given point (line). Try this:_

```
Open #3,ser1: Print #3, "MESSAGE": close #3
```

The quotes are essential. The "MESSAGE" you type in at that point will go to the printer. Try it out of curiosity!

Now here is another crazy thing. I have no idea what useful purpose it could serve, but I will share it with you anyway. No doubt someone has a use for this. It saves a screen to flp or mdv, or whatever, and the details are on page 51 of the Keywords section of the QL Manual. Why not try it out?

```
Sbytes mdv1_screen_data,2^17,2^15
```

("screen_data" being your title or whatever) will save a screen to mdv1. To recover it, try:-

```
Lbytes mdv1_screen_data,(131072)
(Note:- 2^17 = 131072 & 2^15 = 32768)
```

Try putting/typing something on top of the saved screen you have just recovered, save it as above, bring it back to the screen and see what you get.

Found out by happenstance recently, that when writing a letter or article about a program, that if I wanted to include the listing in the article, all I had to do was import it! Here I was running two passes to get it across. To Import the listing to Quill you must have patience, it can take a few minutes. In text87 using the ASCII option, the transfer is almost instantaneous. This is one case where Quill will import a file which is NOT an exp file. text87 will import just about anything. Aint science amazin?

Pass me the aspirin Mabel this thinkin is hurtin, that's right, the brown jug, up on the top shelf---thanks Mabel.

I really do hope I have not bored you. Have fun. I am going back to just sippin. Bye.
SNRRRZZZ §

SOFTWARE REVIEW: OMNICALC 2.

Up to now for my spreadsheet requirements, I have been using Vu-Calc which was the only pertinent program available to me; Vu-Calc has one major shortcoming: it can only handle numbers up to seven characters including a decimal point (that is to say, amounts up to $9999.99 only). Thus a value of $10000.00 cannot be handled. That is a severe limitation in the 1990's.

On seeing a recent catalogue from Jack Dohany (of Mscript fame) which contained a listing "Omnicalc", I wrote to him and asked for a copy. It arrived recently and I went to work trying out this spreadsheet which includes production of histograms (bar charts) based on input data. Jack provided a loader tailored for the Larken system but not for my particular printer, the Fastext 80. I made some changes to the loader to adapt it to my situation. Essentially, I amended it to use the LKDOS printer driver code and decided to use my Graphics 24 code to do the histograms and any other SCREEN$ copies.

There is a very thorough manual provided with the software and the following is a brief summary of some highlights and comparison with Vu-Calc, where appropriate:

1. First and foremost (*): there is no provision for saving or loading data via disk drive. This must all be done on cassette tape. Jack says he might try to modify the code to handle disk drive saves and loads if there is enough interest (ie, if enough people order copies of the program, I would guess). I did however find that an NMI save could do the job quite well, with some reservations. This is done by pressing the Larken NMI button, then pressing the appropriate key (1 fo 4). The program uses 10 tracks but that is no problem. Loading this back merely requires the command <RANDOMIZE USR 100: LOAD "NMI-S1.CM"CODE> for one saved using the <1> key.

Note (*): There is a Spectrum version which allows the use of microdrives and RS232 interface as well as networking. I am not able to investigate these options.

2. My loader allows for the setting of printer parameters (left margin, line length, PICA, ELITE or CONDENSED mode). If the data is loaded via tape, all printing works as expected. However, when loading via NMI, the printer driver must be preset and I did this using my Omnibus <Set Printer> option. Also, if histograms are to be printed, the Graphics 24 code must be preset at this time. Other bit-image copying utilities could be used by doing a SCREEN$ save via NMI first; eg, you could save a SCREEN$ and load it into Pixel Print to use its graphics printing option.

3. I have tested the histogram option and it appears to work quite well. Via tape, there is a choice to save or load either data or workarea. The workarea option allows values to be transferred back and forth between the spreadsheet and the workarea. The latter has many uses, among them being: the plotting of histograms; moving values around within the spreadsheet (ie, columns or rows); communicating between spreadsheets. The last application could be used, for example, to move the closing balance from the end of one year's spreadsheet to the beginning of the next year's spreadsheet. Such transporting of work areas would require their saving and loading via tape until the program is modified for disk drive.

4. Input error detection is provided through different BEEP sounds or error messages; this is quite effective in keeping you on track. A list of error messages is contained in the manual.

5. Values may be expressed in two-decimal-place format or integer format. It's all one or the other, though, unlike Vu-Calc which allows some columns to be displayed in one format and others in a different one. The result is a minor irritation (eg, 180 days shows as 180.00).

6. Depending on the spreadsheet size, calculations are very quick compared with Vu-Calc and saves via tape are quite short. Use of tape is not much of a problem but seems rather archaic. Iterative (ie, repetitive) calculations can be performed and there is much more detail on this in the manual.

7. Very large numbers can be handled but you must reserve TWO columns for large numbers and input them into the right column since display of large numbers is spilled over into the left column. This is a vast improvement over Vu-Calc if you have need for large numbers.

8. Equations are similar to formulas in Vu-Calc but values at the cursor position can be viewed via a separate option <D:Decode>. It also returns codes to signify if a cursor position contains a blank, value, text, row total, row sub-total, column total or column sub-total.

9. The size of a spread sheet can be up to 99 columns across or 250 rows deep; that is significantly more than the 50 by 50 grid of Vu-Calc. All the mathematical functions of the TS2068 can be used making it easy to build up sophisticated models for financial, engineering and other applications which might include tax bands, differential interest rate, conditional expressions and variable overheads.

10. The screen display in the 32-character-per-line standard Sinclair character set shows a window of only three columns and 15 rows at a time plus the column and row headings. A 64 character-per-line font would have been an improvement but you can't have everything. Using the arrow keys to move the cursor is done in two modes: with the Caps Shift pressed, the whole screen is shifted one column or row at a time. With the Caps Shift key not pressed, the cursor moves about the current window. But at least the headings stay on the screen at all times to let you know where you are, unlike Vu-Calc.

11. Text can be entered into any location where the cursor is resting but is limited to all letters (all Capitals), all numbers, all punctuation marks and arithmetic operators which occupy only one character. Only the first seven characters are accepted and spaces count as a character.

12. There is an option <Y> for Year which provides a fast method of feeding in the months of the year, either in columns or rows.

In summary, Omnicalc 2 is recommended to those who are looking for something better than Vu-Calc; there are a few shortcomings but on balance, it does a good job and should satisfy most requirements. It is priced at US$10.00.

Finally, Jack Dohany also provided a Public Notice dated Jan 1, 1991 which in part advises that he is out of the business of supporting the TS2068 and that he will now concentrate on programming and developing hardware for it. Anyone wishing copies of programs he formerly offered should now write to either:

RMG Enterprises
1419 1/2 7th Street
Oregon City
OR 97045

or

Ed Grey Enterprises
PO Box 2186
Inglewood
CA 90305.

Prepared by Bob Mitchell 910121.

TS2068 COMPUTER REPAIRS

Richard Burt wrote to me about his experience in getting his Ts2068 repaired. I think meny members would be interested in hearing of his experience. We have published Dan Elliot's repair list in a recent issue of Sinc Link. Richard's address is:

R.R. #5 Box 157
Belleville, Ont.,
CANADA K8N 4Z5

In your last letter to me you ask for an article that might be of interest to club members, well I don't know if this is an article or a report on a personal experience that happen to me. I will call this "IS THERE LIFE AFTER DEATH". This may sound like a spiritual question, but to any one who has had a Sinclair computer die on them, they may have ask that very question. In Nov. I had one of my TS2068 die on me, it would not initialize, and the screen display was half there. I made a few phone calls to see if anyone could repair this and I was turned down by qualified people citing no repair experience, I was able to buy another backup unit from you. I then gave a call to Dan Elliot of Promise Land Electronics, now called Computer Classic, to see if it could be repaired, and after a conversation with Dan, I decided to send the computer to him. On Dec 10/90 I sent the computer by parcel post, with insurance to Mr. Elliot, and on Jan. 17/91 I received my computer back, that same night I unpacked the unit and hooked it up, and turned it on. The computer initialized and has been working ever since, so the answer to the above question is "YES". I highly reccommend Dan Elliot, for repairs to Sinclair computers, his repair prices are very reasonable, and his work is first rate. I have included a copy of Dan's repair prices and instructions, with this letter.

I now have an extra TS2068 that I am willing to loan out to any club member, in Ontario or Quebec that needs one, while thier's is getting repaired, if they pay the shipping cost both ways, and if they damage it, will agree to repairs to it. I am trying to get another QL for the same purpose. I have a friend that is in need of a 16K or 32K ram pack for a ZX81, if you know anyone that has one for sale let me know. That's it for now.

RJB

ZX81 - TS1000 NEWS

By Jeff Taylor

The biggest thing to hit the ZX81 library in years arrived in the mail last week. No less than 17 High Resolution programs and games! For those users who invested in the Hunter board, the SCRAM board or built the on-board NVM, this is like a dream come true.

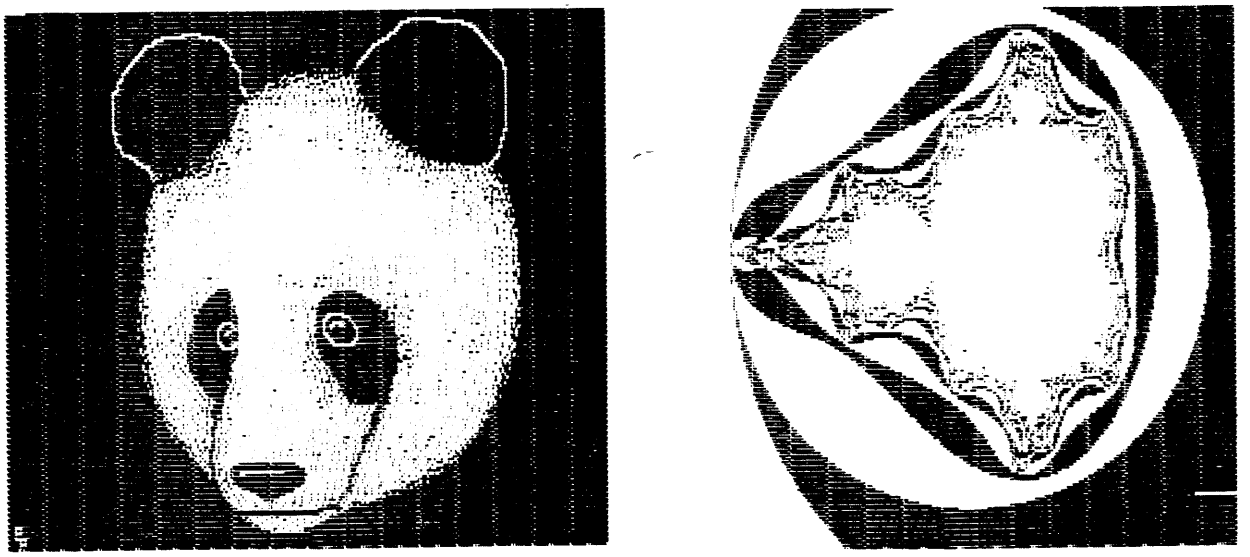
High resolution for the ZX81 you say? Instead of the blocky low resolution (44x64) how would you like to see resolution of 256x176 dots on your screen? With an add-on Hunter-style board or a small on-board circuit, this kind of resolution is available.

Much of the hi-res software developed in North America was done by the team of Greg Harder and Fred Nachbaur (and Wilf Rigter). I contacted Mr. Harder in early February with a request for any information he might still have on the hi-res programs. He was surprised to hear that anyone would still be interested in the old doorstop and graciously offered to copy what he had remaining for us as long as we covered the costs of mailing and tape. Less than three weeks later two tapes chock full of high quality recordings and their manuals arrived. Some of the programs require extra memory (32k+) but several will run with just the 16k Rampack.

The programs range from drawing packages to detailed adventure games, from mathematical art to arcade games. There is even one that offers the option of using a joystick if you have the correct interface.

For more info on hi-res, contact René Bruneau or myself. The tapes will go into the ZX81 tape library.

Does anyone know where Fred Nachbaur is?



ZX81 - TS1000 Hi-Res Printouts

2088 LARKEN DISK MENU MAKER

Larry Crawford 910222

A couple of years ago, George Chambers put together a super program which permitted files on a Larken disk to be loaded with a single keystroke. My kind of utility. I modified it somewhat. With George's permission, here it is.

There are a couple of techniques used which could be useful in other applications, so read on.

The listed program will create an AUTOSTART disk menu on a designated disk. The entry point is line 8888. At this point you are prompted to enter the number of the drive to be processed (dd).

FINDING THE ACTIVE DRIVE

The Larken system stores the active drive number at location 8195 in the LKDOS RAM. To find this number it is necessary to Larken POKE 8200 with 8195. <LET s=USR 110> assigns to s the value stored at that location. This is really a flag with the appropriate bit set. Variable s will have a value of 2, 4, 8, or 16 depending upon which drive is active. The Boolean calculation, <LET sd=INT (s/4-(s>8))>, converts the value of s to an actual drive number. Drive 4, the RAMDISK, has a code of 128 and is not looked after here. However, it would not be used frequently and can be dealt with manually.

In another application, this source drive number could be poked into a safe RAM location (e.g. 26687) for use later on. In this program there is no need to keep the drive number.

ADDING TASWIDE

By using the Taswide utility, it is possible to read a Tasword file directly from the disk without having to load it into Tasword. This is a very useful feature for help files and lengthy descriptions of programs on the disk.

Assuming that is resident, the code "taswi.Cx" is loaded from the source drive and saved to the destination drive.

IF YOU DON'T HAVE TASWIDE

Omit all of line 8888 except for the destination drive input. Also omit lines 3 and 92 and the subroutine 5000 to 5030.

GETTING RID OF UNNECESSARY LINES

Line 8889 deletes line 8888 and itself since they are not needed by the menu program.

AUTOSTART SAVE

Control passes to line 9999, the SAVE line. The variables are cleared out and RAMTOP is poked with 256 bytes more than the value stored in ELIN. The two PEEKS and POKES do this.

Setting RAMTOP this way for the AUTOSTART SAVE allows you to add to the menu without having to worry about exceeding the current value of RAMTOP.

You are prompted to press the <D> key for the save and USR 102 is called to do the job. Control now goes to line 8000.

FINDING THE DISK NAME

In line 8000, thirty-two bytes are poked into the unused area of RAM starting at 24311 (it runs to 24560).

Here is the dissassembly:

```
DI
CALL 0062      (98 dec) :turn on LKDOS
XOR A                   :a=0
LD (201D),A (8221 dec) :current track #
LD (2020),a (8224 dec) :error number=0
CALL 007E     (126 dec) :turn on track 0
CALL 007B     (123 dec) :ld bufr frm dsk
LD HL,3204 (12804 dec) :loc of dsk name
LD DE,6824 (26660 dec) :loc for name
LD BC,0010     (16 dec) :length of name
LDIR                    :move name
LD A (0064)   (100 dec) :turn off LKDOS
EI
RET
```

The USR call to 24311 moves the first 16 characters of the disk name to another unused area of RAM starting at 26660 (it runs to 26687). The program is then RUN.

DATA STATEMENTS

Lines 100 to 1000 are reserved for DATA statements. Each statement must contain two strings of data. The first is a description of the program as it will appear on the screen in the disk menu. The second one is the exact program name as it appears in the disk catalog.

OPERATION

Lines 10 and 11 recover the disk name from its storage area in RAM and assigns it to x$.

Line 20 reads the data lines starting at 100 and prints an identifying letter and the first data statement to the screen via subroutine 7000. The process continues until it hits the dummy statement at 1001.

Lines 50 to 90 prompt you to press a letter key within range and then a loop reads the data lines until the selected one is reached.

Line 91 assigns to t$ the extension symbols of the file name. The file is then LOADed as the appropriate type by line 92, 94, or 98 depending on the contents of t$.

After the program is typed in, SAVE it as "dmm.B1" LINE 8888.

USING THE DISK MENU MAKER

Load "dmm.B1". Place a disk in the destination drive. This disk can be blank or have programs on it but there must be at least 2 free blocks available for the AUTOSTART and another one for Taswide if you are using it.

You will be prompted to enter the number of the destination drive. Once this is entered, "taswi.Cx" will be loaded from the source drive and saved to the destination drive (if applicable).

The tune will play and you will be prompted to press the <D> key. The "blank" disk menu program will now be saved as an AUTOSTART on the disk in the destination drive.

USING THE DISK MENU

After booting the AUTOSTART on the disk in the active drive, <BREAK> in after the menu screen appears. Add the appropriate DATA lines starting at line 100.

A typical line will look like this: <100 DATA "RABBIT'S REVENGE","rr.B1">, the first statement being the description which will appear in the menu and the second the exact name of the file to be loaded.

Then <GO TO 9999> to resave the amended menu as an AUTOSTART.

Whenever a program is added to the disk, repeat this procedure.

The whole process is relatively painless and very useful. I recommend it highly. LDC

LINE DELETE FOR THE SPECTRUM
by G. Chambers

Unlike the 2068 computer, the Spectrum does not have a BLOCK DELETE function. The following m/c program for the Spectrum makes the job of deleting a block of Basic program lines almost instantaneous. To use the facility simply merge lines 9000 to 9090 with the program you want to trim, and type GOTO 9900. Or you could save the machine code itself and then type in lines 9930 to 9980.

The code is relocatable and can be loaded into any other part of RAM by changing line 9905. As it stands CLEAR 31999 should be entered before the program is run.

This utility was lifted from the April '85 issue of Your Computer, page 127. Credit to Rodney Francis

```
8990 REM     DELETE function
             for the Spectrum

9900 INPUT "Address to instal co
de ";c
9910 FOR n=c TO c+52
9920 READ d: POKE n,d
9930 NEXT n
9940 INPUT "First Line ";f
9950 INPUT "Last Line ";l
9960 POKE 23296,f-256*INT (f/256
): POKE 23297,INT (f/256)
9970 POKE 23298,l-256*INT (l/256
): POKE 23299,INT (l/256)
9980 RANDOMIZE USR c
9990 DATA 237,91,0,91,42,2,91,22
9,167,237,82,225,216,122,179,200
,62,39,188,216,32,4,62,15
9995 DATA 189,216,235,213,205,11
0,25,209,192,229,235,205,110,25,
209,192,213,35,35,94,35,86,35,25
,209,205,229,25,201
9998 STOP
9999 PRINT USR 100: SAVE "delete
.B1"
```

SOME POPULAR MISCONCEPTIONS
(Mostly about filenames).

From time to time I read in an article or letter about the QL statements that I know to be untrue or at least misconceived. Sometimes the same misconceptions occur over and over again. The following comments arise from a recent (Jan 90) issue of QL World and the final issue of a US newsletter called Quantum Levels, which started with high hopes as a bimonthly in August 1986 but only managed 12 issues in three years.

Basic Filenames.

Many people seem not to realise that there are two ways of presenting QDOS commands with filenames, and that the rules for allowable characters differ between them. In each case there is a limit of 36 characters in addition to the five which define the device (e.g. mdv2_).

1. The normal method of supplying a "parameter" consisting of unadorned ascii characters when only normal letters, numbers and the underscore character "_" are accepted, e.g. LOAD mdv1_my_prog1.

2. QDOS will also accept strings, otherwise it would be almost impossible to write file handling programs. (I think early versions of the QL only worked this way.) By a string I mean
   i. a set of characters in quotes e.g. SAVE "flp1_!@#$",
   ii. a string variable to which a string of characters has been assigned e.g. COPY a$,b$
   iii. a string function e.g.
MERGE dev$&"_"&pää$&CHR$(233). In these cases the only restriction is that the first five characters must be a legitimate device name followed by an underscore, the rest can be anything you can type in at the keyboard, and even unprintable characters (use CHR$() as above). Thus you could create files with unprintable names that would be almost impossible to delete without reformating the medium - if there were any point to this! Some commercial programmers have used filenames consisting entirely of spaces, which are invisible in a normal listing - I once discovered one on a disk someone sent me that read "FORMAT flp1_:FORMAT flp2_" that was named " "; luckily I discovered it using my Ftidy program before it was activated.

As far as I can see these rules apply to all commands that take filenames in the QL ROMs and in Toolkit II, such as RENAME.

FORMAT also works in the same way as far as medium name is concerned; that's how they got the date in the form 9\9\85 on the early mdv

cartridges.

You can even put one totally nameless file onto each medium; e.g. SAVE flp1_ creates such a file. The file can be BASIC, text or machine code, and is available to any of the commands referring to a single file name. However, DIR, and any commands or programs which make use of the directory, fail to recognise the file at all. If you suspect such a file may be lurking about on one of your disks/cartridges you can reveal it by COPY flp1_,mdv2_secret (or something similar - COPY flp1_,scr_ also works). If a file called "secret" appears on your target device, then it is secret no more and now available to all the normal manipulations.

Psion Filenames.

It is fairly common to read that Quill files must have the extension _doc, export files must have the extension _exp etc., This is not so, these are merely the defaults that the Psion suite uses if provided with nothing else. While you are restricted to a maximum of eight normal letters and numbers for the main part of the name and there must be a thre character extension you can use any three characters that can be obtained from the keyboard for the extension. You must, however, type them all in when you want to save or load the file; "letter_@#$" is quite acceptable as a quill file name for example. You can import any ascii text file into quill too, but it must also fit these rules, so you may have to COPY it to a regulation filename (or rename if you have the facility.) It is the file header itself that distinguishes quill files from abacus files etc. and not the extension as is often thought. I use a three character date code on all my letters e.g. Michael_B12 means a letter to Michael written on November 12th (to me at any rate!)

More Psion tricks

Shift + F5 refreshes the screen in all the programs, but for some reason is undocumented. (Much better than F2 twice!)

Have you discovered that CTRL + down-arrow deletes from the cursor to the end of the line, and CTRL + up-arrow from the cursor to the beginning of the line?

Howard Clase, Tel (709)-753-6415
Box 9947, Station B,
St John's,
Newfoundland,
CANADA, A1A 4L4.

The hardware project presented this month is a parallel interface using the Z80 PIO chip manufactured by Zilog. The interface is fully decoded using Tim Stoddard's design published in Time Designs Magazine for his article "ZX81 Data Acquisition Module" (Real-time Clock and Auxiliary Port) and is compatible with the 2068 and ZX81 data bus at the back of the computer.

The printed circuit provided is laid out to provide a centronics compatible port suitable for driving a big printer without losing the flexibility of providing two input/output ports. The Z80 PIO ports A and B can be programmed as input, output, bi-directional or bit-controlled by sending a control-word to internal registers that configure the ports as required. For use as a printer interface, Port A is configured as input, and Port B is set up as output. We are checking MSCRIPT and TASWORD for compatibility and will report the software modifications required in the May/June newsletter.

Address locations for accessing the ports are as follows:

Port A, Data	53H	83D	
Port B, Data	57H	87D	
Port A, Control	5BH	91D	
Port B, Control	5FH	95D	(H=hex, D=dec)

Programming of the PIO is done by sending a control-word to the correct register.

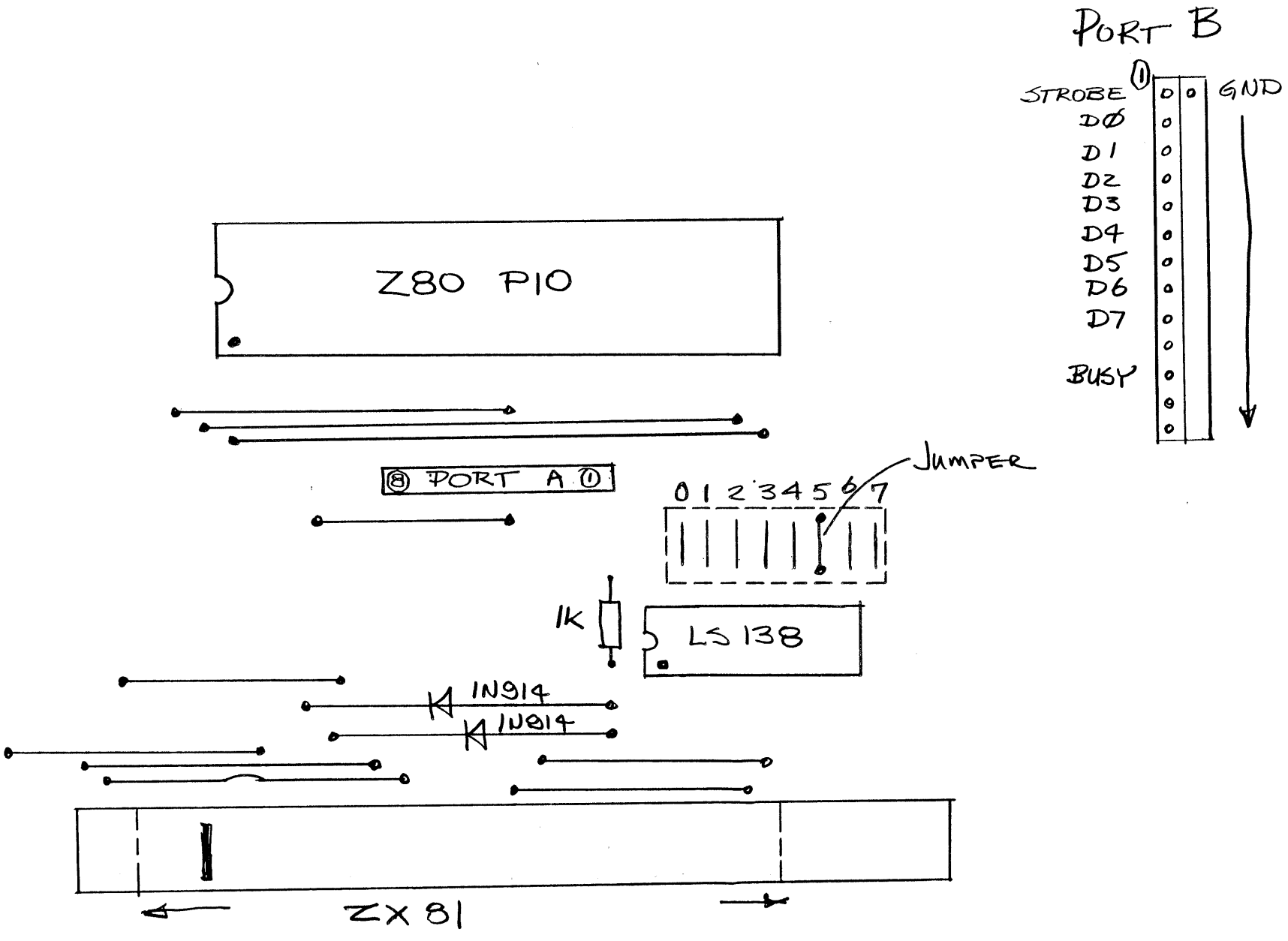
In Hexidecimal	In Decimal	Mode
0F	15	port is output
4F	79	port is input
8F	143	bi-directional
CF	207	bit-controlled

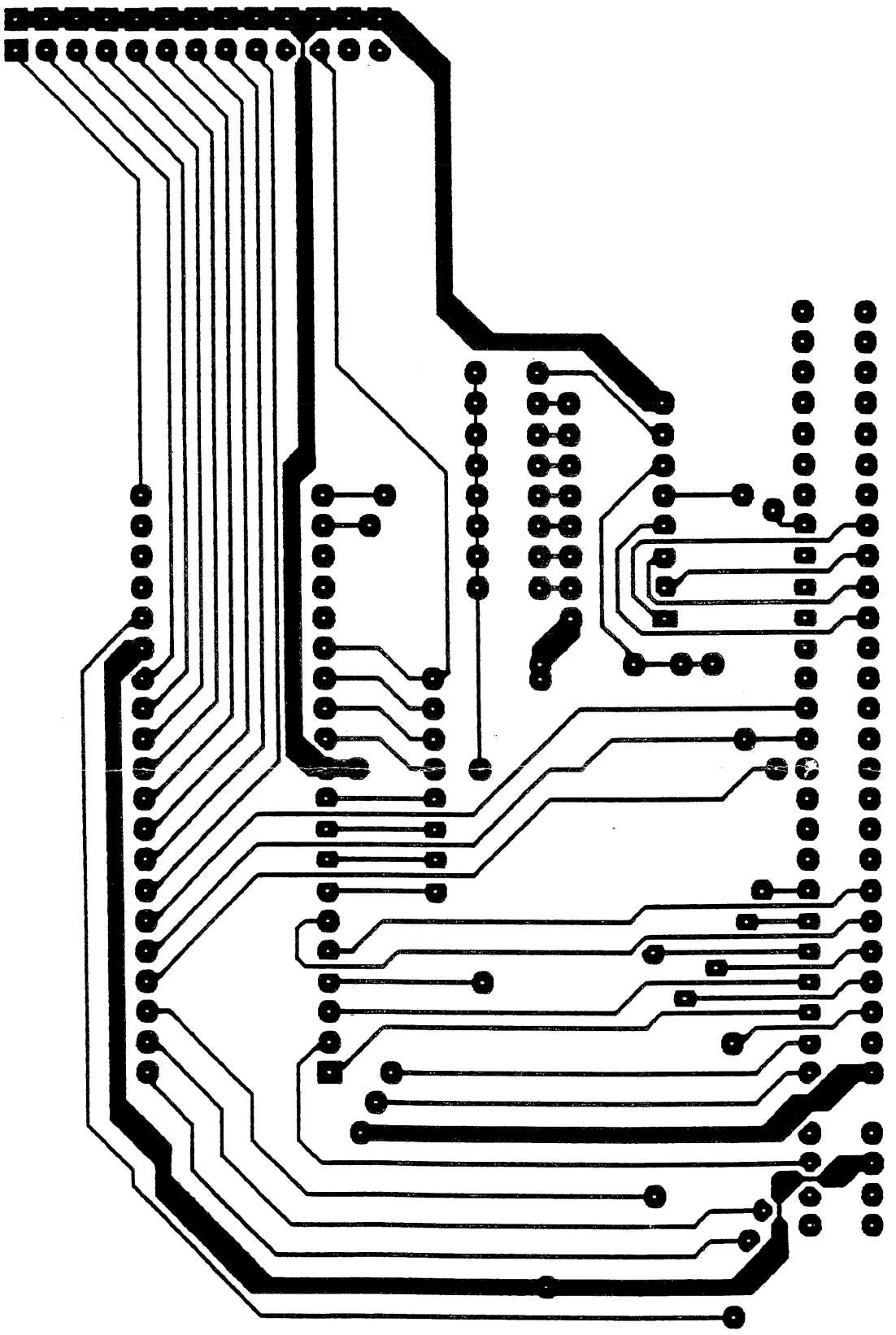
Example: To define Port A as an output in a 2068 system could be done as follows:

```
OUT 91,15
OUT 83,(your data, a number from 0 to 255)
```

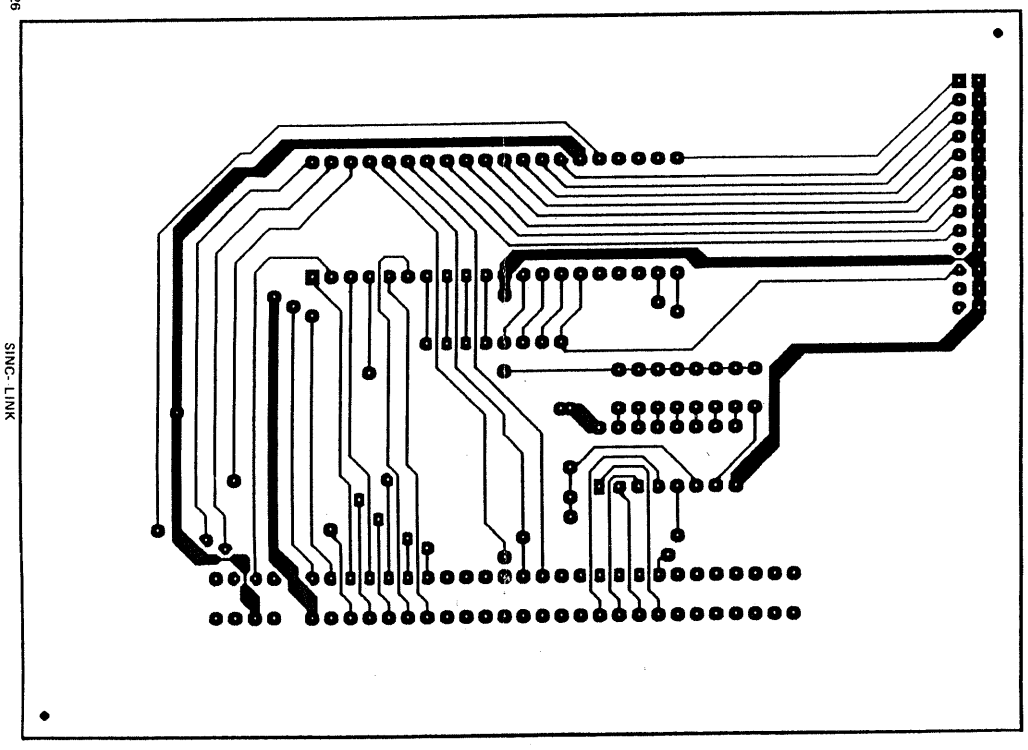
```
001  REM AAAAAAAAA          090  POKE X+7,83
010  LET X=16514            110  PRINT "N=";
020  POKE X,62              120  INPUT N
030  POKE X+1,15            125  IF N<0 THEN  STOP
040  POKE X+2,211           130  PRINT N
050  POKE X+3,91            140  POKE X+5,N
060  POKE X+4,62            150  LET A= USR X
080  POKE X+6,211           160  GOTO 110
```

ZX81 Listing


PORT B
STROBE
GND
D0
D1
D2
D3
D4
D5
D6
D7
BUSY
Z80 PIO
PORT A
JUMPER
0 1 2 3 4 5 6 7
1k
LS 138
IN914
IN914
ZX 81



2x Actual Size From Component Side



2x Actual Size From Component Side

51%

Another Printer Tip
ZX81/TS 1000

Sebastien Boisvert

My recent printer tip articles allowed you to COPY a specific line, COPY a number of lines, so I figured that a routine to COPY a block of lines would also be usefull. First, make a 1 REM with 12 bytes in it and enter the rest of the program. Next, RUN 110 and enter the code at the bottom of the page. Then delete lines 110-140 and there you have it!

```
 10 PRINT "NUMBER OF LINES ?"
 20 INPUT A
 30 POKE 16515,A
 40 PRINT "START LINE ?"
 50 INPUT A
 60 LET A=(A-1)*33
 70 RAND A
 80 POKE 16520,PEEK 16434
 90 POKE 16521,PEEK 16435
100 RAND USR 16514
110 FOR F=16514 TO 16525
120 INPUT A
130 POKE F,A
140 NEXT F
```

The code: 22,0,42,12,64,1,0,0,9,195,110,8

DATA-READ-RESTORE ON A TS1000
by Sebastien Boisvert

In our last issue of Sinc Link (Jan/Feb 1991 Page 8) we ran an article with the above title. Regrettably, we did not include several program listings that went with it.
Here they are:

LISTING 1

```
 1 FOR F=16514 TO 16820
 2 INPUT A
 3 POKE F,A
 4 NEXT F
```

LISTING 2

```
118  118  0    0    0    0    33   132
64   117  35   116  35   64   0    201
42   132  64   58   134  64   167  32
31   229  237  91   12   64   235  237
82   225  40   69   126  254  234  40
3    35   24   237  35   126  254  8
32   243  35   82   1    50   134  64
235  42   16   64   1    6    0    9
235  62   0    50   135  64   126  254
200  40   19   254  118  40   10   237
160  229  33   135  64   52   225  24
237  62   0    50   134  64   35   34
132  64   6    0    33   135  64   78
201  207  23
```

LISTING 3

```
1 CLEAR
2 DIM Z$(32)
3 LET REST=16520
4 LET READ=16530
5 RAND USR REST
```

LISTING 4

```
 1 CLEAR
 2 DIM Z$(32)
 3 LET REST=16520
 4 LET READ=16530
 5 RAND USR REST
10 FOR F=1 TO 3
20 LET A$=Z$( TO USR READ)
30 PRINT A$
40 NEXT F
50 LET A=VAL Z$( TO USR READ)
60 PRINT "RESTORE ADDRESS:";F
70 LET A=VAL Z$( TO USR READ)
80 PRINT "READ ADDRESS:";A
90 PRINT Z$( TO USR READ)
100 PRINT Z$( TO USR READ)
110 REM ▒TS 1000,COMPUTER,DATA
READ/RESTORE PROGRAM,REST,16530
OUT OF DATA ERROR
```

TOS System Repair

by Jeff Taylor

Just after Christmas I had a look at Lyman Paquette's defective TOS disk system. This is the system that Timex of Portugal developed for the Spectrum in the mid-eighties. TOS stands for Timex Operating System. It was sold in the U.S.for the TS2068 (with a Spectrum bus adapter) by English Micro Systems and Zebra Systems and is often referred to as the Zebra disk system.

Anyway, Lyman had borrowed my TOS system to compare with his and had quite rightly determined that the fault lay with the interface module (the unit with the reset button). I offered to have a closer look and try to fix the problem. A little troubleshooting revealed that the fault was one of two chips on a small board which was piggybacked to the main interface board.

"No problem", I thought,"I'll just replace both chips and we'll be back in business". Wrongo, one of the chips had had its identifying numbers ground off. Literally. I could see the marks where someone had actually used a sanding disc on the chip. Why? To prevent the average home handyman from repairing these things instead of sending it back to the dealer, I guess. The trouble is... there are no dealers anymore! My system was no help, the same chips were ground down. What to do?

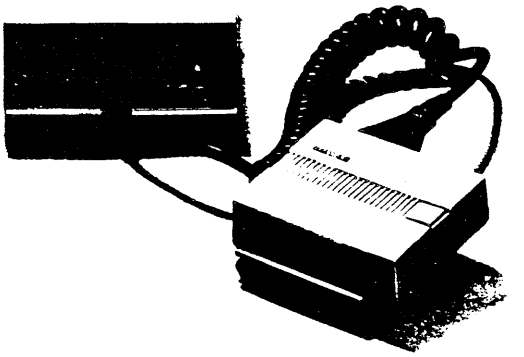
English Micro is long gone but Zebra still exists. I called their number and was told that they no longer supported TS computers. I already knew that but after pressing a little more I was able to get the owner's phone number. Stuart Newfeld (I hope that's spelled correctly) was very helpful. He did not have any replacement parts for the TOS system but he knew who did.

When Zebra got out of the TS market all their TOS parts went to Jack Dohaney (of MSCRIPT fame). A call to Jack in California and a board with the phantom PLA (programmable logic array) chip was in the mail the next day. Jack has also developed a board which combines the interface with the bus adapter in a much smaller and neater package. He also included a tape with a new set of disk utilities and he did all this on credit. He sends the stuff then you pay him. There are too few businessmen like him!

Once the new board was installed properly, the moment of truth had arrived. Power up and presto, the system booted perfectly. Thanks, Jack.

Jack says that in the very near future he will hand all his hardware business over to a dealer to tend so he can devote himself to more programming. He also says that TOS users are not limited to the 3" drives, any drive should work.

My thanks to all the out-of-town members who offered support and even their own systems(!) while I was repairing this one. Is anyone interested in exchanging info about this system? While my Larken system is my favourite, I would like to learn more about TOS, any takers?



Disk drive controller box and plug-in interface

HOW TO SQUISH A DISK

by Bob Mitchell 910112

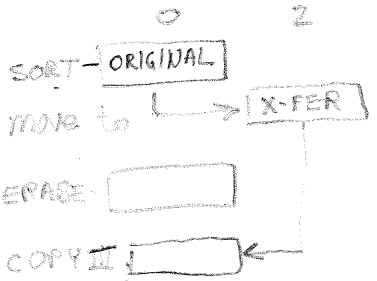
Fragmentation is a phenomenon occurring on disk storage, in which files tend to get broken up into many small extents scattered all over the disk. It is especially prevalent and serious on heavily used disks that have suffered erasures and overwriting.

The rearrangement of data on a disk is called compaction or defragmentation; it is also known as SQUISHING and is usually done by a special utility program. The object of squishing is to accelerate data storage and retrieval thus improving the overall performance of the system. The squished disk will respond much more quickly to header readers like Fullcat and prevent the drive head from moving frantically back and forth searching for the various extents (tracks or blocks in the Larken system; sectors in others) of each file. When a disk is badly fragmented or when a catalogue is sorted, the disk should be compacted.

Compaction must be accomplished (at least for the Larken system) by using a utility that copies the disk file by file , never track by track. Such a file is the Disk File Manager by Jack Dohany which is available from the club library. Another choice would be Richard Hurd's Copymachine.

To squish a disk:
1. Load the copy program (eg, Disk File Manager).
2. Put the disk to be copied into the source drive and a freshly-formatted disk into the target drive.
3. Select the source drive and when the catalogue appears, mark the files with an X by pressing I, then press M to start the MOVE (copy) process. Repeat the process as required to encompass all files on the source disk.
4. Using a disk rename utility, give the target disk its proper name as this was not copied from the source disk.

Terms used above such as squish, compact, fragmented and extent are taken from the following sources:
1. The New American Computer Dictionary by Kent Porter published by Signet 1985.
2. Barron's Dictionary of Computer Terms Second Edition by Downing and Covington published by Barron's 1989.



WORDMASTER TIP

Lionel Keeping sends this tip for Wordmaster on the Larken.

"If you want to save disk space you can save several small files on one track from the file handling menu and thus save some space. For example, my letters to UPDATE magazine are all fairly short but eacch would take up one track if stored separately. If I store several of them in one disk file (still keeping them as separate WORDMASTER files in memory) I can save one or two tracks. It also helps keep track of them all since they are in one disk file. I do the same thing for some of my letters to you, but, as in the case of this one, they usually take up an entire track by themselves."

Lifted from a recent letter to me.
G.F.C.

"LAN" or Local Area Network USING A "QL"

by Hugh H Howie

Do you really know how fortunate you are to own a QL? Do you realise the power and versatility at your finger tips?

In Toronto, we have a monthly paper called TORONTO COMPUTES. This is a free publication, where the Computing Industry can shout its wares. Also included are many excellent articles on the latest and best and worst, in the computer world, in general.

The January issue of this paper has quite an article by David Carter extolling the virtues of LAN, an acronym for Local Area Network. Sound familiar? It should do, as this is what we in the QL world just refer to as "Networking".

I would like to quote some figures from his essay, as they are very interesting. He starts off by saying that 400,000 Canadian micros are already connected with PCLANS, and that the trend is expected to grow to three million by 1993, these figures according to a forecast by IDC Canada Ltd.

He goes in considerable depth into the whys and wherefores of this system, its advantages and its savings. How it is done and its costs. A very enlightening explanation of all it can do, and well worthy of your attention. He makes the subject very interesting, even a couple of diagrams showing a typical set-up with various peripherals added. According to Mr Carter the only costs for peer-to-peer are $100 to $200 for the network interface, (one for each machine) plus the $500 to several thousand dollar cost for cabling and software. If you have to call someone in to install all this, that would cost somewhere between $75 to $200.

This would appear to make the cost to set-up three machines, to be in the region of at LEAST $800. PLUS the cost of the computers. More on this later.

The first in what would appear to be a series on this subject, and from what I have already read, I look forward to the coming essays.

This reminds me of a demo I gave to some QLers a short time ago, (Louis Laferrier wrote about it in the last issue). The demo showed how to send a document from #3 QL to the printer attached to #1. I also had someone play a game on #1, and was able to ask him from #2 to place a program in drive 2, so that I could load it to #2. It was all a matter of criss-cross action and programs and data being transferred back and forth.

To do this did not cost me $800 PLUS the cost of the computers. All I did was use 3 QL's, with the addition of a little gadget called TK2 attached to each, and a cable or two from my stereo, to connect the machines. What did this set-up cost me? Well you all know that the cost of a QL (an orphan computer?) is well under $100, and that the little gadget called TK2 is about $45. That's it, less than $150 for each machine. THAT INCLUDES COMPUTER. A total of $450. Complete. Not $800 PLUS COMPUTERS.

Now I am not going to suggest that the QL system is better than the systems under discussion. All I am saying is that WE ALREADY HAVE IT. Right here in the QL. Admittedly the QL system has its limitations, one is distance, (100 feet) but that can mean across an office or up or down from floor to floor. 10 floors =1000ft?

Why do not more try networking? If you only have one QL, get together with a friend and try it out. All you need is a QL and TK2 at each station, and a cable to connect them. Yes I know added memory is a distinct advantage, but not absolutely necessary. I know that the QL can Network without TK2, but TK2 helps a lot.

The main thing is, we have this at our finger tips, without spending hundreds of dollars. The QL was, and is, far ahead of its time, and does not get the credit it deserves. §§§

=FORMAT TO 894/900 SECTORS? = UH?

Did you ever come across an 80 track disk that looked like the above? This does not mean the disk is no good.

Some software manufacturers format a disk in this manner to ensure no bits and pieces are added to the disk. You can do the same if you have a Cumana I/F, or Trumpcard. With Cumana I/F, use the "flp_opt" command. Type:- "FLP_OPT 2, 30, 50" Then format in the usual manner, and you will have a disk with 900 sectors. (50 tracks x 18 sectors per track). This will only work on 80 track disks. The SPACE between "comma" and the "3" is critical!

With Trumpcard, the command is:- "FLP_Track 30". Once again format in the usual manner, ending up with a disk formatted to 534/540 sectors. This works with 40 or 80 track disks. I have even formatted a disk with 85 tracks (1530 sectors) with this method, but I do not know what effect it would have on data. Strange things perhaps!

Using the Cumana "FLP_OPT 2, 30, 50, the first figure is the Security Level, the second the Startup Time, third is the number of tracks to format to. I would suggest you not play with those first two figures. To regain the original settings, repeat the command using 80 or 40 as desired. Or else RESET. Have fun !

HHH

323 1/2 N. Church Street
Bowling Green, OH 43402
January 26, 1991

Dear George,

I got the latest issue of Sinc-Link today, and am glad to say you and the club seem to be thriving.

I read with interest Bob Swoger and Larry Kenny's article on changing the stepping rate on a Larken disk. Perhaps that is one problem I have had with attempting to use disks between my two drives, though I have not attempted to verify it yet.

However, I noticed a serious mistake in the program. In line 80, the new stepping rate is POKEd to 40015, but the READ/POKE in line 120 writes over this! In other words, the only rate this program will change to is the one in the DATA, which is the 30ms rate. The slow rate will not prevent someone with a faster drive from reading the disk, but means the program does not actually do what it says.

The cure is trivial: change line 80 to line 130. Then the POKE will occur after the DATA has been READ, and all will be well.

Note also that those values are for the old controller. The same numbers will have different meanings if you have one of the newer controllers. My system with an FD1797 from AERCO uses the same stepping rates, so I can't guess what the new rates might be, though 1, 2, 3, and 6 might be a good guess. If you have one of the new controllers and an appropriate drive, it really won't matter since it will still be slow enough.

Also, Réal Gagnon's Sinclair Trivia is missing one answer, the one about the unique feature of the 2068. Since he said 2068 and not Spectrum, I can only presume that it is a feature the 2068 has but the Spectrum does not: either bank switching, the AY3-8912 sound chip, or the fact it has 2 joystick ports.

I see you also published my Tasword file compressor. I hope no one tries to write me at the address given, though, since that is my home address. Well, I suppose my parents would forward it here.

A suggestion for for a more elegant way to handle the OUT and delay on page 17 (or anywhere else). We could make something which acts like an LPRINT to some extent. Also, the delay may need to check a different number on different printers, as you well know (when I changed from a Fastext 80 to my current Panasonic, I did have to change numbers).

The routine would work as follows: place the item(s) to be sent to the printer into a string (I will use p$). Then call the following subroutine:

```
1000 LET p=IN 127: IF p-8*INT (p
/8)+16*INT (p/16)=229 THEN GOTO
1000
1005 IF LEN p$=0 THEN RETURN
1010 OUT 127,CODE p$: LET p$=p$(
2 TO)
1015 GOTO 1000
```

The string for the original program would be as follows:
LET p$=CHR$ 27+"3"+CHR$ 15+CHR$ 15 +CHR$ 27+"T0": GOSUB 1000.
To me at least, this is somewhat more elegant than the original. Also, it will let you "LPRINT" any string without needing to set up the LKDOS OPEN #3,"lp" command, but you will need to add an ENTER to each line (CHR$ 13).

For the record, there are at least 4 signals from the printer which the AERCO interface passes to the computer by IN 127. There are actually 5 lines from the printer to the computer which are supposed to carry signals, which are labelled in my printer manual as ACK, PO, BUSY, SLCT, and ERROR (the Fastext 80 uses some slightly different names). The nearest estimates I have for the different bits are as follows:

Bit 0 Not used (always high)
Bit 1 PO Paper Out
Bit 2 ERROR (high means no error)
Bit 3 ??? unknown (possibly SELECT or ACK)
Bit 4 BUSY (high means buffer full or printer off-line)
Bits 5-7 not used (always high)

The primary difference between my printer and yours as far as IN 127 is concerned is bit 3 above - yours appears to be always high, while mine stays low. Hence, you get 237, but I find 229 when I "PRINT IN 127". The line 1000 in the routine above removes only bit 3, thus changing your 237 to my 229 but leaving mine alone. You could use line 1000 anywhere you normally use IF IN 127<>237 THEN...

Speaking of which, is the 108 you listed an error? It does not agree with your previously listed value of 237.

Note the 27, 51, 15 is equivalent to 27, 65, 5 if your printer has superscripts but not n/216 inch spacing - some of the older EPSON and compatible printers don't. Also, people with IBM Proprinter compatible printers will need to add an additional 27, 50. I notice your printer could not handle the subscript command. The small print of the subscript looks very good in compressed pitch, and changing the spacing makes it look almost like you simply reduced it to about 60% of the original. Using compressed and reducing the print spacing to 8 lines per inch is not as good looking, but will do in a pinch.

Oh, if I got a red face every time I made a mistake, people would think I had a constant sunburn. In my line, you discover mistakes are unavoidable, and try to correct them as quickly as possible. So you will have to forgive me my bad habit of jumping on any error I see, as I expect people to jump on my errors.

Here is a list of the various values I get from my printer for various printer conditions, along with my interpretations thereof:

Code	Printer State	Interpretation
225	Printer Off	All lines low
229	Printer Ready	Line 2 high (no error)
231	Paper Out/Online	Lines 1 and 2 (paper out, no error)
241	Offline	Line 4 high (Busy and error)

243 Paper Out/Offline Lines 1 and 4 (Busy, error, and PO)
245 Printer Busy Lines 2 and 4 (Busy and no error)

Presumably, your printer may have 8 added to most of these. Additionally, I believe the Fastext 80 does not possess a non-error paper-out condition. Therefore, you would not have a 231 or equivalent. There are other differences, particularly in the SELECT line, but from these values I suspect that line is not available. I still don't know what line 3 is, which your printer has high and mine low. It isn't the SELECT line, but perhaps it is some line not used on either.

I did not get to mail this last week - I misplaced the copy I had printed, and am just about to re-print it. Take care, and I will write you again later. Peace!

Sincerely,
Steven V. Cumbow

P.S. At the top of page 2, the line reading LET p$ = CHR$(27) +... ends in a "TØ" (i.e., number zero, not letter O). While it is easy to see the difference on the screen, with the slashed zero, most printers do no such thing. Sorry for any confusion.

CLEAN SCREEN

I came across this 'letter to the editor' in the Aug '85 issue of YOUR COMPUTER. This question and answer may be of interest to many of our members. It also works on the TS2068. GFC

"I have written several games for the Spectrum, in which I load a complete picture (saved as Screen$) into the computer, so it can sit there while the main program is loading. However, I don't want the loading message of the second program to obliterate parts of the picture which is on the screen. I know I can overcome this, to some extent, by controlling the PAPER and INK colours of the loading message, but this puts some restrictions on the original Screen$ picture. Is there a better way?

A. Steve Dempster, of Littleworth, Stafford, has discovered a very good way of solving this problem. To prevent the filename from obstructing your Screen$, first PEEK 23570. Note down the answer you get. Next, after the Screen$ has loaded, have your loading routine POKE 23570,16. After the main program has loaded, POKE 23570 with the original number.

```
  5 REM  Listing entered and
 modified for the Larken System
          by G. Chambers.

 10 POKE 23658,0: CLS : PRINT A
T 2,10;"IMAGE EDIT"'''"Your Compu
ter April 1985, p 135";'''TAB 6;
"Menu:"''"1/2/3/4   Load Screen"
''"S          Start Editor"''"Q
      Quit"
 15 PRINT '"6          Return to
 Basic"''"0          Clear Screen
"''"8/9       OVER/OVERlap Black
"
 17 PRINT '"H          Help"
 20 IF INKEY$="q" THEN  STOP
 30 IF INKEY$="s" THEN  GO TO 1
00
 35 IF INKEY$="h" THEN  GO TO 1
000
 41 LET c=CODE INKEY$: IF c>48
AND c<53 THEN  LET sc=(CODE INKE
Y$)-49: INPUT "Input Full Screen
 Name ";n$: RANDOMIZE USR 100: L
OAD n$CODE sc*6144+40000: GO TO
10
 50 GO TO 20
 100 CLS : RANDOMIZE USR 23296:
PRINT #0;"Save Image Y/N ? "
 110 IF INKEY$="y" THEN  INPUT "
Name ? ";n$: RANDOMIZE USR 100:
SAVE n$SCREEN$ : GO TO 10
 120 IF INKEY$="n" THEN  GO TO 1
0
 130 GO TO 110
1000 CLS : PRINT TAB 12;"IMAGE E
DIT"
1010 PRINT '" Have you ever wish
ed to combineindividual screen$.
 This is     difficult in Basic
as you cannotmerge two screens o
r load one   with OVER  on."
1020 PRINT '"  This program, Ima
ge Edit, letsyou combine picture
s."
1040 PRINT '"   You will be pres
ented with a menu. If you want t
o combine 2, 3, or 4 images, loa
d them into  different SCREEN$.
Press keys 1 to 4 in turn to loa
d the desiredscreens into memory
. When you   have loaded all the
 desired     screens, press the
S key to     start the machine c
ode program."
1050 PAUSE 0: CLS : PRINT "   Th
ere are two different ways of co
mbining images, using the  m/c c
ommands OR or XOR."
1060 PRINT '" 'XOR' puts the ima
ges together the same way as the
 Basic       command OVER. Where
 there is an overlap, a space is
 made. 'OR'  adds the images com
pletely,     leaving the overlap
 black. When using the m/c progr
am, press 1, 2, 3, or 4 to put a
n image on   the screen. It will
 be added to whatever is on the
screen       according to the mo
de XOR or OR."
1070 PRINT '"  When returning to
 the program you will be given t
he opportuni-ty of Saving the co
mposite or   loading more images
."
1080 PAUSE 0: CLS : PRINT '''"  I
t is possible to create some oth
er effects by combinations ofthe
 two modes. For example, you can
 subtract an image from what is
already on the screen by     add
ing it first in OR mode, thenin
XOR mode."
1090 PRINT '"  When combining pi
ctures,      colours are ignored
 as they     would be jumbled wh
en combined.   The final image i
s Saved as   black on white"
1100 PRINT 'TAB 6;"**************
**"
1110 PAUSE 0: GO TO 10
8999 STOP
9000 RANDOMIZE USR 100: SAVE "im
age.B1" LINE 9020
9010 STOP
9020 BORDER 7: PAPER 7: INK 0: C
LEAR 39999: RANDOMIZE USR 100: L
OAD "image.C1"CODE 23296: GO TO
10
```

```
start   23296 5B00 LD B,247;
        23298 5B02 LD C,254;
        23300 5B04 LD A,B
        23301 5B05 IN A,(C)
        23303 5B07 AND 1;
        23305 5B09 JR Z,23360   one
        23307 5B0B LD A,B
        23308 5B0C IN A,(C)
        23310 5B0E AND 2;
        23312 5B10 JR Z,23383   two
        23314 5B12 LD A,B
        23315 5B13 IN A,(C)
        23317 5B15 AND 4;
        23319 5B17 JR Z,23406   three
        23321 5B19 LD A,B
        23322 5B1A IN A,(C)
        23324 5B1C AND 8;
        23326 5B1E JR Z,23430   four
        23328 5B20 LD B,239;
        23330 5B22 LD A,B
        23331 5B23 IN A,(C)
        23333 5B25 AND 16;
        23335 5B27 RET Z
        23336 5B28 LD A,B
        23337 5B29 IN A,(C)
        23339 5B2B AND 4;
        23341 5B2D JR Z,23468
        23343 5B2F LD A,B
        23344 5B30 IN A,(C)
        23346 5B32 AND 2;
        23348 5B34 JR Z,23476
        23350 5B36 LD A,B
        23351 5B37 IN A,(C)
        23353 5B39 AND 1;
        23355 5B3B CALL Z,2214  (clear screen)
        23358 5B3E JR 23296

one     23360 5B40 LD DE,16384
        23363 5B43 LD HL,40000
        23366 5B46 LD BC,6144
        23369 5B49 LD A,(DE)
        23370 5B4A CALL 23454   subroutine
        23373 5B4D LD (DE),A
        23374 5B4E INC DE
        23375 5B4F INC HL
        23376 5B50 DEC BC
        23377 5B51 LD A,B
        23378 5B52 OR C
        23379 5B53 JR NZ,23369
        23381 5B55 JR 23296     start

two     23383 5B57 LD DE,16384
        23386 5B5A LD HL,46144
        23389 5B5D LD BC,6144
        23392 5B60 LD A,(DE)
        23393 5B61 CALL 23454   subroutine
        23396 5B64 LD (DE),A
        23397 5B65 INC DE
        23398 5B66 INC HL
        23399 5B67 DEC BC
        23400 5B68 LD A,B
        23401 5B69 OR C
        23402 5B6A JR NZ,23392
        23404 5B6C JR 23296     start

three   23406 5B6E LD DE,16384
        23409 5B71 LD HL,52288
        23412 5B74 LD BC,6144
        23415 5B77 LD A,(DE)
        23416 5B78 CALL 23454   subroutine
        23419 5B7B LD (DE),A
        23420 5B7C INC DE
        23421 5B7D INC HL
        23422 5B7E DEC BC
        23423 5B7F LD A,B
        23424 5B80 OR C
        23425 5B81 JR NZ,23415
        23427 5B83 JP 23296     start

four    23430 5B86 LD DE,16384
        23433 5B89 LD HL,58432
        23436 5B8C LD BC,6144
        23439 5B8F LD A,(DE)
        23440 5B90 CALL 23454   subroutine
        23443 5B93 LD (DE),A
        23444 5B94 INC DE
        23445 5B95 INC HL
        23446 5B96 DEC BC
        23447 5B97 LD A,B
        23448 5B98 OR C
        23449 5B99 JR NZ,23439
        23451 5B9B JP 23296     start

subroutine 23454 5B9E PUSH AF
        23455 5B9F LD A,(23681)
        23458 5BA2 BIT 0,A
        23460 5BA4 JR Z,23465   2-sub
        23462 5BA6 POP AF
        23463 5BA7 OR (HL)
        23464 5BA8 RET

2-sub   23465 5BA9 POP AF
        23466 5BAA XOR (HL)
        23467 5BAB RET

        23468 5BAC LD A,0;
        23470 5BAE LD (23681),A
        23473 5BB1 JP 23296

        23476 5BB4 LD A,1;
        23478 5BB6 LD (23681),A
        23481 5BB9 JP 23296

        23484 5BBC NOP
        23485 5BBD NOP
```

23296 - PRINTER BUFFER.
23681 - UNUSED BYTE IN SYSTEM VARS

image.C1

```
100 REM       IMAGE EDIT
          The code for the
          Image Edit program

105 REM  ** For the TS2068**
       To use on the Spectrum
POKE 23356,107; POKE 23357,13

110 RESTORE 120: FOR n=23296 TO
23486: READ a: POKE n,a: NEXT n

120 DATA 6,247,14,254,120,237
121 DATA 120,230,1,40,53,120
122 DATA 237,120,230,2,40,69
123 DATA 120,237,120,230,4,40
124 DATA 85,120,237,120,230,8
125 DATA 40,102,6,239,120,237
126 DATA 120,230,16,200,120,237

127 DATA 120,230,4,40,125,120
128 DATA 237,120,230,2,40,126
129 DATA 120,237,120,230,1,204
130 DATA 166,8,24,192,17,0
131 DATA 64,33,64,156,1,0
132 DATA 24,26,205,158,91,18
133 DATA 19,35,11,120,177,32
134 DATA 244,24,169,17,0,64
135 DATA 33,64,180,1,0,24
136 DATA 26,205,158,91,18,19
137 DATA 35,11,120,177,32,244
138 DATA 24,146,17,0,64,33
139 DATA 64,204,1,0,24,26
140 DATA 205,158,91,18,19,35
141 DATA 11,120,177,32,244,195
142 DATA 0,91,17,0,64,33
143 DATA 64,228,1,0,24,26
144 DATA 205,158,91,18,19,35
145 DATA 11,120,177,32,244,195
146 DATA 0,91,245,58,129,92
147 DATA 203,71,40,3,241,182
148 DATA 201,241,174,201,62,0
149 DATA 50,129,92,195,0,91
150 DATA 62,1,50,129,92,195
151 DATA 0,91,0,0,0,0
 310 PRINT AT 15,5;"Data has bee
n entered"'!" Press a key to se
lect drive,          and save to
disk."
 320 PAUSE 0: INPUT "Drive # ";d
rv: RANDOMIZE USR 100: GO TO drv
: RANDOMIZE USR 100: SAVE "image
.C1"CODE 23296,188
8990 STOP
9000 RANDOMIZE USR 100: SAVE "im
age.Bx"
```

ALFA-BYTES #1

by the Tonkin Reader

There has recently been a lot of discussion about what will and will not be compatible carry-over programs from the Spectrum to the TS2068. The determining factor seems to lie in the exclusive use of fully documented ROM calls. As any good programmer knows, Basic, which we find in ROM, is an understandable combination of assembly language instructions called MNEMONICS, some of the most common of which are JNP, BRA, STA, BCC, LEA, ETC (et cetera). The following are a few of the lesser known instructions that are present in most, if not all computers.

ARG:Agree to Run Garbage
(definitely the most often used euphemism among users)
DDS:Damage Disk and Stop
(who hasn't seen this one?)
VMB:Verify and make bad
(precedes ARG in operation)
SDJ:Send all Data to Japan
(ever wonder where it went?)
TTC:Tangle Tape and Crash
(created just for CCR users)
YII:Yield to Irresistible Impulse
(occurs after VMB/ARG; results in purchase of new case)
KFP:Kindle Fire in Printer
(need that report when?)
LNM:Launch Nuclear Missiles
(you were on WHAT BBS?)

It's pretty certain that most of us have experienced at least one of the above, and this is by no means an exhaustive listing. Watch next time as ALFA-BYTES brings you additional tid-BYTES from the Tonkin Reader.

from Nite Times News

I have some SINCLAIR magazines for sale. Shipping is included with the price quoted for each package.

Sinclair Magazine all originals, no copies
TS HORIZONS (USA,2068,ZX81) no. 1,2,3,4,5,6 _____ $ 4.00

BASIC COMPUTER NEWSLETTER (USA,zx81) vol 1 no. 2,3,4,5 ____ $ 2.00

SINCLAIR PROJECT (british, hardware oriented.ZX81.spectrum)
dec/jan 83 to aug/sept 84 (10 issues)_____ $ 7.50

SINCLAIR PROGRAMS (british, 40 programs to type in in each issue, ZX81.spectrum)
jan/feb 83, may 84, june 83 _____ $ 3.00

PERSONAL COMPUTER (italian magazine, excellent coverage for Spectrum and QL)
no. 1 to 19 _____ $20.00

SINCLAIR USER (big british magazine, spectrum, few QL)
jan 85 to dec 85 _____ $15.00

SINCLAIR USER (big british magazine, spectrum)
april 83, may 83, jul 84 to dec 84 _____ $12.00

SINCLAIR USER (big british magazine, spectrum, few ql)
may 86 to nov 86 _____ $12.00

QL USER (british, QL only)
feb 85, apr 85 to nov 85 _____ $15.00

UPDATE (USA, 2068 and QL)
oct 88, jan 89 to oct 90 (total 9 nos.) _____ $10.00

PERSONAL COMPUTER WEEKLY (british, excellent coverage for spectrum and QL, weekly magazine)
vol. 4 no. 25 to 51, except 34 (total 26 issues) ___ $25.00

PERSONAL COMPUTER WEEKLY (british, excellent coverage for spectrum and QL,weekly magazine)
vol. 5 no. 1 to 36 and 38 (total 37 issues) _____ $35.00

SUM (USA, 2068, few QL)
vol. 3 no.6 to 9, vol. 4 no.1 to 6 (10 issues) _____ $ 6.00

ZX COMPUTING (british, spectrum and QL)
apr/may 84, apr 86 to jul 86, sep 86 to dec 86
jan 87 and mar 87 to jun 87 (13 issues) _____ $25.00

TIMEX-SINCLAIR USER (USA, ZX81)
vol. 1 nos. 1,2,3,4,6 _____ $ 5.00

SYNC (USA, ZX81)
vol. 2 no.3-6, vol. 3 no.1-2 _____ $ 4.00

SINCLAIR QL WORLD (british, QL only)
dec86, jan87 to feb87, jun86 to nov86 (9 issues)___ $15.00

Réal Gagnon
8286 St-Hubert
Montréal (Québec)
H2P 1Z3
514-381-6264

Jan 15, 1991

Dear Out-of-Town Members,

I'm not sure that I can fill a page this time. Maybe it's because of the Christmas season, I don't know.

But I'll start off with a plea from our newsletter editor. He has a Portugeuse disk system designed for the 2068. It was marketed by English Microconnection and/or the Zebra people. Jeff asked me if any of the OOT members might have such a system. If so, would you look at the innards of the Interface box. That is the one with the reset button on it. THere are two chips in this box, and on both Jeff's and another system he has looked at the code markings on these two chips have been carefully scraped off. Jeff would like to know what what these chips are. The part number is sufficient, if it is showing. Though, I suppose that if anyone had the schematic for this system, it would show this information.

From looking at a recent issue of TIMELINEZ, published by American Micro Systems, I see that they are going to discontinue publication. All TIMELINEX subscribers are going to have their subscription filled by receiving issues of the SNUG newsletter, ROUNDUP.

In this issue of TIMELINEZ, Jul-Dec 1990, there is an advert by the CAMBRIDGE GROUP, inviting club members to become what they term "part-time sales consultants", to sell the Z88 computer on campus at colleges and universities. No pricing is given. They give a phone number to call, 800-458 9008. Ask about becoming a Cambridge Colleague. Caution though; I just noticed that this advert, in the form of a letter, has a date of January 19, 1990; a year old.

Also in this issue is a series of documents put out by the District of Maine Bankruptcy Court. They pertain to an agreement reached between SSI Computer Systems, Inc., and a couple of banks that are owed money. It appears there is an agreement between the parties to let SSI continue to sell their product contingent on remitting $150 per item to the banks. What is the product, you ask? Well none other than the Z88 computer. Seems they have about 700 Z88's to sell, plus another 273 refurbished units, and 345 failed units.

There is also a phrase "d/b/a CAMBRIDGE NORTH AMERICA", in one of the documents; it is unclear to me just what the connection is.

While we're on the subject, there is an advert in the INDIANA T/S UG newsletter. It is by MACro Systems and offers a 2-pound computer called the "Personal Assistant". Special price for a 32K system, $325 US; for a 2 Meg system, $1225. You can also buy additional RAM cartridges; starting at 128K for $75, up to 1024K for $495. EPROM cartridges are said to be coming 1st quarter of 1991 (haven't I heard that before?), for about the same pricing. If you haven't guessed, a squiggle at the bottom of the page says this is the Z88. MACroSystems E.T.C., 1207 South State Street, Salt Lake City, Utah 84111. tel. 801-575 8855.

Anybody want an AMDEK disk drive system. Two 3" drives along with the power supply, all contained in a case. Along with about twenty 3-inch disks. This belongs to a former member, Greg Lloyd, who used it on his Larken system. They happen to be at my place. I have mentioned them before in the newsletter, but they are still around. I think I told Greg they were worth more than what it turned out Timex owners wished to pay!! Well, that's how it goes. Actually I think the disks would be a good buy for anyone who already has such a system. You get a goodly supply of disks plus a backup to your disk system. Why don't I use it? Well, I have enough complications already, with DS and QD drives!! I used to have a system like this, and I must say, it worked beautifully. Drop me a line if you are interested.

Got my income tax forms today, and so I took a look at them to see what changes would be necessary to my income tax program. There are quite a few minor changes. The program is only useful for Canadians, and then it is oriented more to Ontario taxpayers, since it includes the Provincial income tax. However, if anyone is interested in, ask.

I don't think there has been any additions to the Larken library. We are up to disk #40, if that says anything. I think that's where we were a couple of months ago. The tape libraries are pretty static these days also. I think there is probably more programs in those libraries than anyone can cope with, so there is not much urgency to adding to them. I can't tell you much about the QL library; you'll have to ask the QL librarian.

Sincerely, George Chambers