Still Alive With Sir Clive!

ZXir QLive Alive!

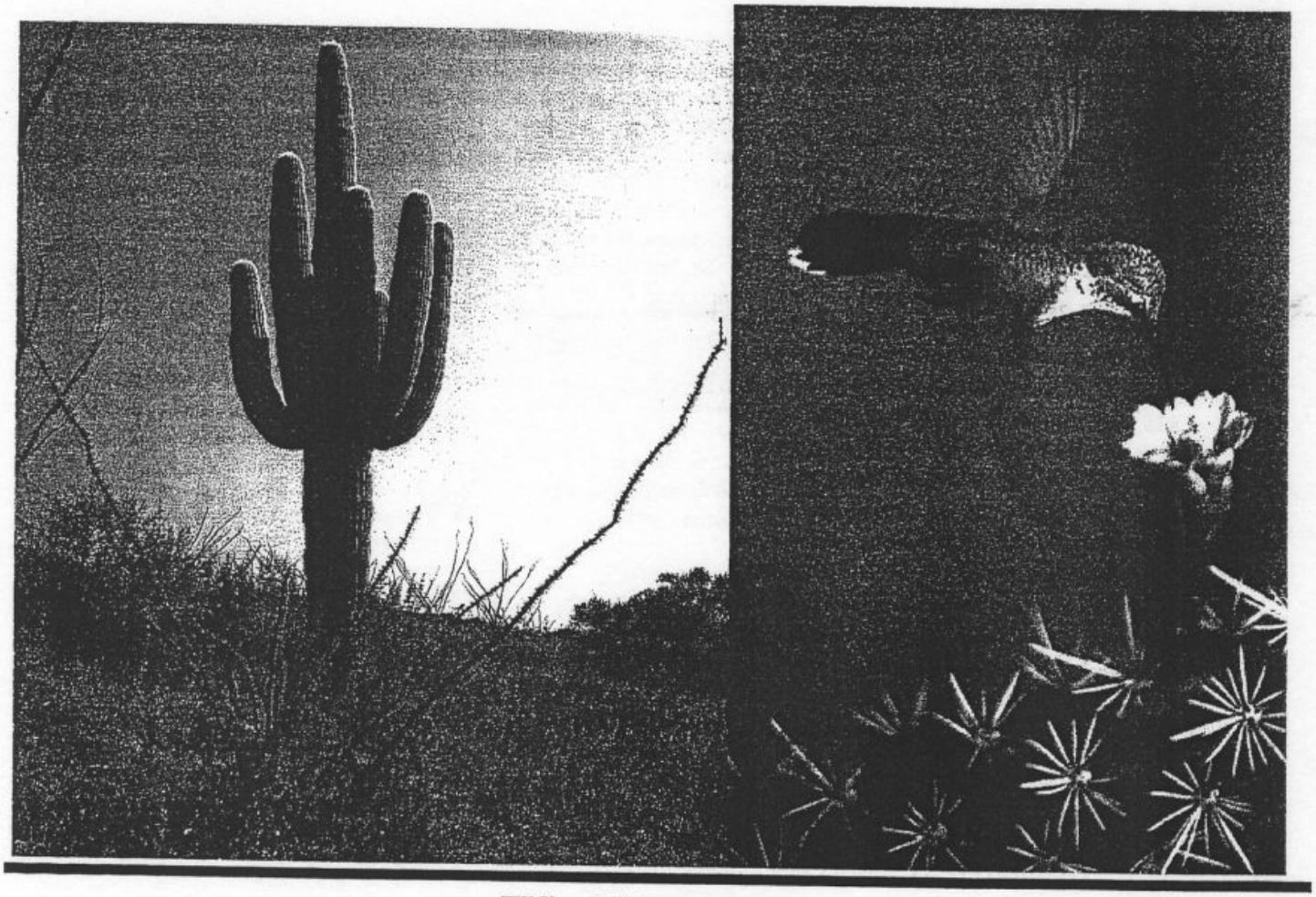
The Timex/Sinclair NorthAmerican User Groups Newsletter

Volume 9 No. 2 Summer '99

MEMORY MAP

ADDRESS	ROUTINES
2	Information and Chairmen — Trea$ury Note$
3	*Input/Output — by Abed Kahale*
3	Larry Kenny (LarKen)
3	**Sir Clive Sinclair** speaks
3	**CTM** ?
4	**More Software** – Jack Boatwright
4	Rod Gowen E-mail
4	Sinclair E-Mail List
5	**QL Today** Notes - Robert Hartung
6	From the Chairman's Disk - Donald Lambert

ADDRESS	FILES
5	**QL Today** Notes - Robert Hartung
7	**Font Loader** – HiSOFT Pascal – David Solly
8	**Save** on Ink-Jet Ink
8	**West Coast Sinclair Show** – Tim Swensen
9	**TS-2068 ROM** Bypass I - WIDJUP
11	ZX-TEAM
12	How to Hack on the ZX Spectrum — Les Cottrell
	SUBROUTINES
23	Unclassified **Ads**



ZXir QLive Alive! ©

Established 1991 The Timex/Sinclair NorthAmerican User Groups Newsletter

T/SNUG Information

We wish to support the following platforms : ZX-80/81, TS-1000, Spectrum, TS-2068, Z88 and QL. If you have any questions about any of these fine Sinclairs, contact the:

Chairman

Chief Motivator
Donald S. Lambert (ISTUG)
1301 Kiblinger Pl.
Auburn, IN 46706-3010
219 925-1372

Vice-Chairmen

Tape & JLO PD Library
D. G. Smith
415 Stone St.
Johnstown, PA 15906
814 535-6998

Z88 Library
Dave Bennett (HATSUG)
1275 Timber View Dr.
Mechanicsburg, PA 17055-9146
717 732-4374

QL Hacker's Journal
Timothy Swenson
2455 Medallion Dr.
Union City, CA 94587-1914
swensontc@geocities.com

TS-2068
Rod Humphreys (VSUG)
10984 Collins Pl.
Delta, BC V4C 7E6 Canada
604 583-2819

QL PD Library
John Donaldson (CATUG)
835 Foxwood Cir.
Geneva, IL 60134-1631
630 232-6147

AERCO & Z80 Emulator
Keith Watson
41634 Amberly Dr.
Mt. Clemens, MI 48038

BBS --==GATOR==--
Bob Swoger (CATUG)
613 Parkside Cir.
Streamwood, IL 60107-1647
630 837-7957 Work 847 576-8068

Any of the above can also be reached by E-Mail through the **Club BBS 847 632-5558**

ZXir QLive Alive!

Is the newsletter of T/SNUG, the Timex/Sinclair North American User Groups, providing news and software support to the T/S community in a **VOLUME** of four newsletters per year; beginning with the Spring (March) issue.

T/SNUG's main goal is to preserve and encourage the use of Sinclair computers by providing an open forum for the exchange of knowledge, building and maintaining of software libraries. Providing vendors, repair service and members with free ad space.

It is the user groups and individual subscribers, rather than the vendors, that provide the pecuniary support for this newsletter. Vendors and developers receive this newsletter free of charge, though contribution from vendors and user groups is gratefully accepted. Please support our vendors and service providers whenever possible.

If you have a problem or you have solved a problem, please share it with the rest of us. No problem will be considered unimportant.

Editor/Treasurer Publisher

You can keep T/SNUG alive by an annual contribution of $12 for one VOLUME made payable to Abed Kahale. Send check to:-

ABED KAHALE
3343 S FLAT ROCK CT
SIERRA VISTA AZ 85650-6874

520 378-3424

Back copies are available for $1.00 each postpaid.

Treasury Notes

As of June 29, 1999, we have a balance of $880.22

Article Contributions

Send in your articles by disk, hardcopy or mail, e-mail and your inputs to:—

Abed Kahale
E-mail: AKahale@compuserve.com

WELCOME

Glen Goodwin

GATOR's

Twisted Pair

To better inform the Sinclair Community, three 24-hour a day BBSs are now provided to serve you. You are encouraged to exchange mail and use the files sections of these boards. Bulletins and ads are available to all.

Q-Box BBS 810 254-9878
Utica, Michigan
SOL BBS 520 882-0388
Tucson, Arizona
Club BBS 847 632-5558
Arlington Heights, Illinois

WebPages
http://users.aol.clubbbs/tsnug/
http://www.outlawnet.com/~jboatno4

If you know the Internet E-Mail address of a Sinclair user, but do not have access to Internet, simply address your E-Mail to GATOR Sinclair on the 24-hour **Club BBS** and include the name and E-Mail address of the user you wish to reach. Then check the Club BBS from time to time if you expect a reply.

We encourage you to exchange mail and contribute to the UPLOAD section. Call and ***register*** using your first, last name and phone number along with a password you won't forget. ***Write It Down!*** Do not try to do anything else at this time.

When you call-in the next time, you will have Level 5 security and be able to enjoy full user privileges. The BBS has smaller sections called conferences. Select "J " for "Join a Conference". Select "TIMEX" to get into the Sinclair Section. The mail you then read will only be from other T/S users. Use extension .ART for articles, .ADS for ads and .NWS for news when UPLOADing.

For help, contact the SYSOP, Bob Swoger, by leaving a message, mail, E-Mail or phone.
CENG108@email.mot.com

Input/Output

by Abed Kahale

Hello Abed....

I was talking to Larry Kenny recently and he tells me that he is willing to release all his information on LarKen DDIs to the public domain. All he is asking for is a working Timex/Sinclair 2068 and the last version of his T/S 2068 DDI in return. (It seems that both were "lost" during a move.) This is very little to ask for this information and the possibility that a LarKen disk access could be built into one of the existing Sinclair emulators such as Gerton Lunter's Z80 emulator or the Warajevo emulator. (Much of my best T/S 2068 software is locked up on LarKen formatted diskettes as NMI saves that can not be accessed by any other means.)

Perhaps the best way for anyone who wishes to make an offer of either one would be to write Larry Kenny directly. His e-mail address is **larken@storm.ca**. Just remind him what the offer is in aid of. There is also a LarKen Electronics web site.

Also, for the benefit of those who subscribe to ZQA! but not to the list, you could put an announcement to this effect in the next publication.

David Solly

Hello Larry,

I just got a letter forwarded to me from T/SNUG by Abed Kahale from Dave Solly.

I am Bob Swoger, --==GATOR==-- of GATOR Software Development - the author of LogiCall for the LarKen Disk Interface on the TS2068.

Next time you are surfing the INTERNET check out:

http://members.aol.com/clubbbs/ (to see my picture - ugliest thing you'll ever see)

http://members.aol.com/clubbbs/tsnug (where all North American Sinclair happens)

http://members.aol.com/clubbbs/catug (our little Chicago Club)

Let me know what you think about our homepages ---

--==GATOR==--

Hello Abed....

Program listings are best done in a fixed pitch (or mono spaced) font like OCR-A or Courier. This is especially true of QL BASIC and Pascal where the indentation is an indication of program flow which is useful for debugging.

People should also be encouraged to use a font in which a 1 (one), does not look like an l (el), or a 0 (zero) does not look like an O (oh) and so on. It helps reduce the frustration levels of those who are trying to type in and run programs from listings.

I think, in the long run, this will also save you a lot of work as well as making ZXir QLive Alive! a consistently high quality newsletter. (Sometimes the paralinguistics don't come though in the written form of the language. I hope this is coming through as positive suggestions. :))

David Solly

Mr. Swoger:

I'm a recently revived ZX-81 user and happened to stumble onto the T/SNUG site. I noticed you have a page with many items for sale at very reasonable prices. Whom should I contact about purchasing some of these items?

Also please forward T/SNUG membership information as I wish to support groups such as yours who keep the Sinclair line of computers alive. Regards,

Glen Goodwin

Hi Abed,

Here's another news article I found on the 'net about the next Sinclair machine.

Jack Boatwright

Linux Tempts Sinclair Back Inventor Promises to Undercut PC Market With Linux Machine.

Inventor and entrepreneur Sir Clive Sinclair is planning a return to the IT market with a low-price portable machine based on Linux and non-Intel chips, writes Robert Juman Blincoe.

Sinclair believes the product will receive support from the corporate and academic markets because Linux has already secured great loyalty in these sectors.

He claims his proposed machine, which will take two years to come to market, will be built around a cheaper processor than Intel's.

"The standard PC is expensive because of Intel and the software, which is demanding of memory," Sinclair said. "The reason why my machine will be cheaper is that it will use a lot less memory, a lower-cost processor, a simpler power supply and a lower-cost operating system."

Sinclair says his new machine will be released at less than half the price of other similar-sized PCs on the market. He supports the loyal Linux users who want Windows to be unbundled from PCs.

"There should be one price for a machine with Microsoft and one price for a machine with Linux," he said.

"Linux looks like a way in - a Trojan horse," Sinclair continued; "a lot of software suppliers are now supporting it. They wouldn't do that if they didn't have a lot of confidence in it. I think a dedicated Linux machine will be the next step."

22 April 1999 Business Publications Ltd.

Dear Abed,

Peter Liebert-Adelt purchased some T/S magazines from me and because a couple of **CTM** mags were in those that I sent ... Peter asked if I could give him any "fill in" about them? Apparently they were devoted to computers and radios? I know nothing about the mag. I wondered if you (or perhaps; one of your readers) might be able to help Peter? Sinclairly,

Fred Henn

Dear Abed

I recently snail-mailed you a memo requesting a copy of DUS 5.0. I did get a copy and thank you for your help.

As you can see; I have finally made the jump to the

p.c.- windows system.

And as you can imagine I have far too many text files to consider printing and then re entering them into the windows format.

The question then is; Is there a dos/windows utility that will allow a p.c. to read TS2068 LarKen text files?

I do not plan to sell the TS2068; but for the sake of convenience would like to be able to read the LarKen diskettes with this p.c. Thank You

Don Oviatt c.d.oviatt@worldnet.att.net

As far as I know, the only way to read TS2068 files in a PC is through a modem, if you have a TS2050 modem that is.

One way is to UPLOAD a file from the 2068 to a BBS for instance and then DOWNLOAD that file by a PC.

Or connect the PC and the 2068 modems together and have one computer UPLOAD or DOWNLOAD to the other computer. A little flaky but I have done it in the past. But I will steer you to Robert Swoger (Chicago Area Timex User Group) clubbbs@aol.com

---

..... I also have some software items for TS2068 that I have gotten in recent purchases. **Profile 2068, Tasword 2** and some others. If you want a list I will try to get it before the next edition. I also have **TS2068 cartridges; Crazybugs and States Capitals.**

Jack Boatwright

Date: Thu, 3 Jun 1999 12:19:38 -0400 (EDT)
From: jl911@cleveland.Freenet.Edu (Rod Gowen)
Subject: Thanks for ZQA!
Hello Abed,

Well, I am finally able to do email. I cannot send or receive attachments, but I can use the mail! I received the disk last week with the I/O file on it. I must say that it was in sorry condition. The metal slide cover was so bad that I had to remove it even to get it into a drive. I did manage to get it in, load the file and then I threw it away. Thank you for the file.

Now, however, you can put whatever you have in or can put into ASCII text format and send it to me as an email letter. It will save a lot of trouble on your part.

Jack Boatwright has been keeping me up to date until now. Now I am able to get on his site and jump to other TS pages from there. It is interesting. I do not use windows to do this. I am using a DOS program called Commo V7 and my local library has a Telnet connection to the internet. I can go anywhere and even download files, I just cannot attach things to the mail I send. That is one of the restrictions of the :freenet: concept.

Well, I will quit for now. Hope to hear from you when the new ZQA! is ready. Later,

Rod Gowen

To Bob Swoger Larry Sauter Cy Herre John Donaldson and Phillip Kiwtkowski Thank you very much for the flowers sent to my wife's funeral.

Sinclair E-Mail List

Albrecht, Alvin	aralbrec@concentric.net
Anderson, Paul	p.aderson@cwix.com
Anson, Gerald	jerrya@aztec.asu.edu
Barker Robin	Robin@di-ren.demon.co.uk
Bennett, Dave	dbennett10@desupernet.net
Boatwright, Jack	jboatno4@outlawnet.com
Boehm, Al	boehm@ziplink.net
Boehm, Bill	boehm@plh.af.mil
C. A. T. S.	mf0002@epfl2.epflbalto.org
Cable, Bill	bcable@triton.coat.com
Catotti, Christopher	kd4ace@compuserve.com
Chambers, George	gfchamb@pathcom.com
Collins, Bill	bcollins@home.ifx.net
Cottrell, Les	jacottre@gte.net
Cruz-Figueroa, Jaime	cruzfiguer@aol.com
Dansby, Andrew	adansby@atlantic.net
Davis, Frank	fdavis@iquest.net
Dunnet, Ron	ron@qubbesoft.freeserve.co.uk
England, William	wengland@iname.com
Feng, Al	alfeng@juno.com
Fink, Mike	domino.cubes@excelsior.net
Fink, Mike	domino.cubes@pointblank.com
Firshman, Tony	tony@firshman.demon.co.uk
Florit, Louis	florit@wormhole.unixville.com
Franke, John	j.m.franke@larc.nasa.gov
Ganger, Gary	gangerg@dma.org
Gillespie, Doug	aa431@cleveland.freenet.edu
Girnius, William	girnius_w@bls.gov
Gowen, Rod	jl911@kanga.ins.cwru.Edu
Harbit, Ken	krh03@cvip.fresno.com
Henderlight, Mike	mikehend@microsoft.com
Henn, Fred	oranur@juno.com
Humphreys, Rod	rodh@lightspeed.bc.ca
Hunkins, James	jdhunki@ibm.net
Impellizerri, John	jimpellizerri@compuserve.com
Jaap, Matthias	matthias_Jaap@hhs.hh.schule.de
Jonas, Mike	mjonas@bbn.com
Jones, Dilwyn	dilwyn.jones@dj.softnet.co.uk
Jones, Terry	tjones@iname.com
Kaczor, Jon	75363.1127@compuserve.com
Kahale, Abed	akahale@compuserve.com
Kealy, Harriet Joan	hjkealy@admin.hilconet.com
Kenny, Larry	larken@storm.ca
Kingsley, Ed	edk4@aol.com
König, Urs	urs.koenig@agrodata.ch
KurtK7	kurtk7@aol.com
Kwitkowski, Phillip	kwit47@aol.com
Lancaster, Garry	dharkhig@delphi.com
Lanciault, Francois	francois.lanciault@energies.alstom.ca
Lassov, David	emanon@azstarnet.com
LaVerne, Melvin	mlaverne@usit.net
Lebowitz, Dave	dkl@dpliv.com
Lessenberry, Gary	gl743@aol.com
Liebert-Adelt, Peter	p.liebert@t-online.de
Malloy, Bob	74776.2342@compuserve.com
McKelvey, William	mckelveyw@delphi.com

Merz, Jochen	jmerz@t-online.de
Merz, Jochen	jochenmerz@j-m-s.com
Miller, Seymour	seymil@delphi.com
Muth, Bob	bobkeeper1@aol.com
Norton, Gary	gnorton@world.std.com
Parrish, Gil	gil.parrish@abanet.org
Pashtoon, Nazir	nazir.pashtoon@ingram.micro.com
Paul Holmgren	paulholm@indy.net
Payne, Josh	joshpayne@bigfoot.com
Pazmino, John	john.pazmino@moondog.com
Perry, Russ Jr	slapdash@enteract.com
Rigter, Wilf	rigter@cafe.net
Rish John	74601.1535@compuserve.com
Shepard, Jay	jshepard@netins.net
Simon, Thomas	73177.333@compuserve.com
Skapinski, Tom	tskapins@juno.com
Smith, Dennis	denny.smith@juno.com
Solly, David	ac355@freenet.carleton.ca
Stegman, Dan	danesteg@juno.com
Swenson, Tim	swensontc@geocities.com
Swenson, Tim	swensont@sirclive.csd.sgi.com
Swentko, Wally	wswentko@maroon.tc.umn.edu
Swoger, Robert	ceng108@email.mot.com
Taylor, Jeff	jetaylor@mdrobotics.ca
Thoresen, Jeff	74200.257@compuserve.com
Walterman, Don	walterm@ix.netcom.com
Watson, Keith	Keith_watson@juno.com
Wood, Roy	qbranch@qbranch.demon.co.uk

QL Today Notes

by Robert Hartung

The QL Today news notes which I send to the ZQA! Editor from time to time are obviously reports on reports and reviews that appear in recent issues of that Newsletter. I don't intend this to be like some of the "talking heads" on TV today who seem to be only reporting on the opinions of other reporters instead of researching the actual news sources. What I'm trying to say here is that I hope these brief reports on QL Today news items will stir up interest among some QL users so they will want to "go to the source" and subscribe to QL Today themselves. It should go without saying that only if there is continued support for those who are supplying information, hardware, software, and services will these things continue to be available, here, as well as overseas.

One factor some state as a reason/excuse for not buying their own subscription to QL Today (presently DM75 or £30) is the difficulty of converting currencies -- no problem when plastic is used. The first of the year marked the introduction of the Euro dollar -- sort of. For a time it will be used only for electronic transfers and exchanges, not as actual currency. Also, for the time being the UK has opted out of participation with the eleven members of the European community who are on board, so the British pound is still "the coin of the realm" in all transactions made there. Apparently, all this is giving headaches among European QL users as well as those here in making currency conversions, so an Italian users group has created a neat little QLiberator-compiled routine that only requires entering either the value of the Euros or any of the eleven local currencies in order to convert either way it runs fine under QPC and SMSQ/E. It even has a multi-tasking sleep mode that is toggled on/off by CTRL C.

This routine is included among the programs and listings that are on the bonus QL Today Cover Disk that comes with each final volume-issue of the newsletter, VOL. 3, No. 6 in this case. All the published listings, an annual index of articles, and several of the freeware programs reviewed during the preceding year are also on the 3½" floppy in zip-compressed files, as well as the UNZIP program.

The current Cover Disk includes an update (v.TUTK3c27) of the code module for TurboToolkit, along with many demo plug-in procedures which utilize this toolkit. Also on the disk is a full version of Mark Knight's K-BASE (v.2.20), which utilizes Turbo Toolkit and was compiled under the Turbo compiler. This multi-tasking card-index style database program is designed principally for quick and easy access to records such as address lists, inventories, etc. and will store and process up to 32767 records. It has a comprehensive and clearly-written manual, stored in ASCII, and has received very positive reviews. It runs fine under QXL or QPC and SMSQ/E as well as Trump Card or Gold Card expansions of the QL.

A new upgrade of the Perfection wordprocessor is likely to be released as freeware, but beta-testing of it was not yet completed at the time the current Cover Disk was made up. one of the features of this version is that high resolution screens are used properly, so you may pre-configure large windows if you wish.

In other news, Richard Zidlicky, the author of uQLx emulator for Unix/Linus based machines, has ported Linux-68k to the newly-released Q40 (the hardware replacement for the QL, with a fast 68040 processor). This raises the interesting possibility that the Q40 itself might be able to run a QL emulator under Linux. Richard is now looking to have full Q40 support added to future releases of the Linux system. (Linux, so named in 1991 for a Finnish university student, Linus Torvalds, who has developed it from its roots in Unix to its present kernel version 2.2.0, runs on all major processors, and has become a formidable competitor to Microsoft Windows and NT.) (and is Free) This, in turn, may have the potential of creating a new market for the Q40 among Linux users that would enlarge its production base among them as well as providing QL users with a full-fledged platform for the powerful Linux OS. This would provide the best of both worlds, as well as further insurance of continued availability and improvement of QL hardware and software! Since the UK does not yet use the Euro, I wrote the following little SuperBasic routine that converts pounds to US dollars. The PRINT_USING command to format the output requires that TK2 be activated, as is automatically done under QPC.

```
1 REMark QPC ED Listing As Displayed in
WTV Mode
```

```
10 CLS
20 PRINT "British pound £ to US $
conversion"
30 INPUT \\"Enter current $n.nnnn to £
rate: $";pr
40 INPUT \\"Currency in £s to convert:
£";np
50 PRINT \\"US dollars: ";:
PRINT_USING"$$###.##",pr*np
60 PRINT \\"Any key to repeat - CTRL
SPACE to halt"\"then RUN to change RATE"
70 PAUSE 1000
80 CLS
90 PRINT "Current $ to £ rate: $";pr
100 GO TO 40
```

FROM THE CHAIRMAN'S DISK

Donald S. Lambert

Under the column Article Contributions on page 2 there is a box in which it states that "Send in your articles by tape or disk and your inputs to me" that has to be changed. Sometime between this newsletter, Summer '99 and Fall '99, I will be down to one computer and that will be a Z88. By then I will have sent all my other computer equipment to Jack Boatwright. I don't really want to part with my 2068 BUT! wifely pressure and a move is forcing the issue. With that said I will continue with this column

Way back when I bought a Z88 EPROM Eraser that requires an AC voltage of 220 volts. Someone responded with the information that a supplier had a 110 VAC to 220 VAC plug in transformer so I ordered it and it is a cute little thing. But before I got the transformer I had bought a power supply that had the ability to runoff of 110 VAC or 220 VAC. So I had converted it so that it supplied 220 VAC. But that thing was a heavy and big deal. While it is not according to electrical code, I had ran the 220 VAC to a standard type 110 VAC socket and put a standard 110 VAC plug on the line cord of the Z88 EPROM Eraser. Today as I was cleaning and sorting things I came across the EPROM Eraser and I dug out the step up transformer. I stared at the socket on the transformer and I noted that it was for a round U.K. type plug prongs. BUT then I noted that it had also the provision for the flat plugs like on the US 110 VAC line cords. Yep it fits I presume it works since I did not test it.

I thought that a 2068 on the other computer desk had gotten contrary but today while copying a 3.5 720K disk to DSDD 360K disk I discovered that drive 0 (full height 360K) drive must be bad since drive 3 (half height 360K) drive worked all right. That 2068 will be in the next box of TS goodies that I will ship to Jack I have a full box to send out as soon as I seal it up, put the shipping label on and take it to UPS for shipment

Now the '**news**' we have bought a house in Forsyth, IL. And where is Forsyth IL? I you take I-72 into Decatur, IL and exit going south you are in Decatur but if you exit going north your are in Forsyth. Decatur is not growing and housing is rather limited I think there are new houses but not located where it is easy to get to the shopping. BUT! the only enclosed mall and a lot of the stores for the area are in Forsyth. They are building all over in Forsyth. Our oldest daughter and family live in Decatur and are about 7 miles from where we are buying. There are grandchildren there and the youngest is in 7th grade going into 8th next year. Looking back we have averaged bout 10 years in a house since we started buying houses

We went to Decatur to see the granddaughter in show choir of the intermediate school. It was a long performance since they put on all their competition stuff. We will be going back Memorial day weekend to attend the grandson's high school graduation party and also to meet with the builder on the 1st of June (our wedding anniversary, 48th) to pick out the flooring and such for the house. We picked out the siding colors and brick work and interior paint on May 4th and the kitchen cabinets May 5th. How did we find the house. Well, we were there the week end of April 23rd and we arrived early so we went driving around Forsyth and spotted this house with workers working on the roof putting on the sheathing. They invited us to take a look at the rough framing of the house and they gave us a set of blue prints. We sort of dallied along till I put the pressure on for my wife to make up her mind. So when I called on May 3rd to talk to the builder we found that the house was quite a few thousand less than we had been quoted as a possible price by the workers.

The builder's brother was in another city 45 miles away talking to a prospective buyer and since the financing was too iffy for that party we had a verbal agreement to buy the house. After we had gotten the house on the evening of the 4th of May as I was relaxing they showed the tornado damage in Oklahoma. Yes! we will have a basement! I looked on the map and Gil Parrish is not too far east of that area I don't think he got any damage.

I realize that this is not too much about TSing but it will explain my lack of input and my address change. I have an address but I will hold that off to reconfirm it and find out more about the move with a tentative date of August 1st this year. Right now we are in the throes of getting the house here ready to sell and packing, sorting and disposing of extra stuff or no longer wanted stuff.

It is about 310 miles one way and it seems longer especially with I-74 having a lot of construction west of Indianapolis to the IL border. 0/0

* * *

The armed attack on that high school would not have happened if the other kids had been armed. When the gunmen attacked the classes all the other kids could have drawn their pocket protection and blown them away. This just reinforces the NRA's message that none of us are safe until everybody and that includes every single child is adequately armed.

* * *

Font Loader in HiSOFT Pascal©

Article and Program by David Solly

One of the attractions of the Timex/Sinclair 2068 is its ability to change its screen font through the use of user-defined fonts. As a programmer, you may want to change the display font a program uses for several reasons. A banking or financial program, for instance, would require that you use a font in which no one can possibly mistake a letter like "el" for the number one. To better set the mood of a game set the Middle Ages, you may wish to use a Gothic font. Likewise, a game set in outer space may require a futuristic looking font. Perhaps you want your T/S 2068 to display text in Greek or Russian. In that case, you have to change the entire character set as well as the font. (To the confusion of many, computer programmers and technical writers tend to use "font" to mean either a "font" or a "character set". Be aware that the program I am about to describe can be used to change both.)

Font loading and activation is much easier in HiSoft Pascal, (henceforth: Pascal), than in BASIC for the following reasons:

Firstly, you don't have to make any changes to RAM_TOP. To prevent BASIC programs from overwriting a user defined font, the usual practice is to lower RAM_TOP and load the font into the protected area created above. In Pascal, a global array ÿ which is also a static array ÿ is used to reserve space in RAM into which the font is loaded. Pascal keeps track of such arrays and prevents them from being overwritten by any of its operations.

Secondly, you don't have to worry about where in RAM the font file is loaded. The built-in function ADDR(v) can be used anywhere in a program to locate the start position of any variable the program uses. Even if you should make modifications to your source code which changes the location of a given variable within the compiled program, ADDR(v) will still be able to locate it.

Finally, you are relieved of doing and re-doing the calculations needed to determine which values to poke into the system variable CHARS to activate the new font. Once ADDR(v) has been used to locate the start of the variable being used to store the font, all your program is required to do is to invoke the POKE(l, n) procedure to transfer this value less 256 bytes to CHARS, thus: POKE(23606, ADDR(v) - 256); and your new font becomes active.

There are already font creation programs aplenty written in BASIC. Most of them save the fonts they create as a byte file of 770 bytes. Below is a demonstration program written in Pascal that contains source code which will allow you to load and activate these fonts within a Pascal program.

```
PROGRAM FONTLOADER;
{
  Special note to HiSoft Pascal 1.7M
users:
   This program must be compiled in 31
   column mode.
}
CONST
  Chars = 23606;
   {Chars holds the address of the
   system
   variable CHARS which in turn points
   to
   the location either in ROM or RAM
   where
   the current font is located.
   The value stored in CHARS is always
   256
   bytes less than the actual start
   location of the current font.}
VAR
  CharSet : ARRAY [0..770] OF CHAR;
   {Reserve space in RAM for the new
   character set}
   {Remember:  CHAR takes 1 byte,
   INTEGER takes 2 bytes}
  FileName : ARRAY [1..12] OF CHAR;
    {An array for holding the file
     name of the font to be loaded}
  StartAddr, {To hold the start address
of the new font}
  I          {A loop counter}
    : INTEGER;

BEGIN   {MAIN PROGRAM}
PAGE; {Clear the screen}
WRITELN('What is the name');
WRITELN('of the font to load?');
WRITELN('(12 characters padded)');
WRITELN;
WRITE('> ');
READLN; {Required in HS Pascal}
READ(FileName);
{Load in the font}
TIN(FileName, ADDR(CharSet));
StartAddr := ADDR(CharSet);
   {Poke location of new font
    into the system variable CHARS}
POKE(Chars, StartAddr - 256);
   {Print the new font from space to
   UDG-U}
FOR I := 32 TO 164 DO
BEGIN
  WRITE(CHR(I), ' ');
END;

WRITELN; WRITELN;
{Add font save code here if required}
WRITELN('End of test');
END.
```

A special note for ZX Microdrive users:

For some reason this Pascal font loader will only load fonts that have been saved on tape. Trying to load font file which has been saved to the Microdrive from BASIC will

generate an "invalid file format" error report. If you wish to use the Microdrive to load in a new font, you can, but first you must create a special kind of font file from within a Pascal program for each font and save it on the Microdrive.

First, add the following lines to the source code in the spot indicated in the main program and resave. Then recompile and save the new font loader/saver program.

```
{Save font routine}
    WRITELN('Enter a name to save');
    WRITELN('the font to the
    microdrive?');
    WRITELN('(12 characters padded)');
    WRITELN;
    WRITE('> ');
    READLN; {Required in HS Pascal}
    READ(FileName);
       {Save the font}
    TOUT(FileName, ADDR(CharSet),
    SIZE(CharSet));
```

Next, if you do not have your character sets saved on tape already then using BASIC you will have to load in each font from the Microdrive in and resave it on tape. (I trust that you all know how to do this.)

Once you have your font files saved on tape then you can use the Pascal font loader/saver program to load them back from tape and save them out ÿ and here I would strongly suggest using a new cartridge ÿ to the Microdrive. Now you character sets are ready for you any time you wish to use them in a Pascal program.

The source code for this program is available upon request by sending e-mail to David Solly at ac355@ncf.ca. Source code is saved as an *.MDX file which is compatible with Gerton Lunter's Z80 ZX Spectrum emulator.

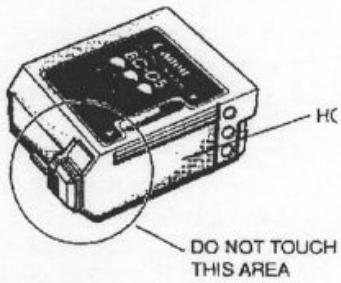
BEATING THE HIGH PRICE OF PRINTER INK

Editor

So you got a good deal on that ink-jet printer only to find out later that the cost of buying new cartridges is breaking the bank. Each cartridge costing between $18 to $ 34 and producing at most 300 pages. It is easy to spend more on ink during your first year of ownership than you paid for the printer itself. Suddenly that $ 99.95 ~ $ 149.95 color ink jet doesn't look like such a great bargain.

You can slash the ink cost radically by purchasing refill kits from various suppliers who sell those kits for various ink-jet (bubble-jet) printers like Cannon, Epson, HP, Lexmark etc.... for around $27 per ink bottle that is good for 14 re-fills in my case.

My kit included .8 oz. of ink, a syringe and a little hand drill with directions. You poke a hole in the cartridge with the little drill, then using the syringe, you inject the water soluable ink into the cartridge – doing that over the sink or a newspaper – just in case!!! It is almost like getting ink for free.



Voila! you are back in business for a fraction of a penny per page.

Nu-Kote International
800 448-1422 www.nukote.com
Renewable Resources
800 734-6548 www.renewableresources.com
Repeat-O-Type
800 228-3330 www.repeatotype.com

West Coast Sinclair Show

by Tim Swenson, the show organizer

The West Coast Sinclair Show was held in Union City, CA, on 5 June 1999, one week after the East Coast Sinclair Show. Most of the European attendees arrived the Monday before the show and spent the week visiting San Francisco.

The day before the show was by Bar-B-Q at my house. Before the show a trip was made to a local regional park to show the Europeans a little bit of California nature. Luckily the side trip was enjoyed by all.

The attendees at the Bar-B-Q were: Simon Goodwin and his girlfriend Chris Lyle, Tony Firshman, Roy Wood, Jochen Merz, Marcel Kilgus, Dietrich and Inga Bruder, Jim Hunkins, Don Walterman, John Rish and Jack Boatwright. As evening fell, Simon, Chris, Tony, Marcel, and I hopped in the Hot Tub for a bit of a soak.

The day of the show I loaded up the station wagon QL stuff, picked up a few folks at the local Motel and headed for the venue.

As this was the first show I've organized, I was not too sure on how big a place to get. The venue as, shall we say, nice and cozy. There was enough room for the vendors, but not really enough for a lot of milling about.

The vendors were - Tony Firshman with his ever present QL controlled LEGO robot, Roy Wood demoing software on a Q40. Jochen Merz and Marcel sharing a table with their wares. Jack Boatwright was selling the last of the stuff from RMG. John Rish (the sole US QL, Z88 dealer) had mostly Z88 stuff on his table. Don Walterman had a Spectrum 128 set up for all to see. Simon Goodwin tinking on a QL (one of two) that had been given as orphans looking for a new home, by a QLer who had upgraded to Mac.

Ken Harbit drove from Fresno to pick up some T/S 2068 stuff and went home with one of the orphaned QLs. Bill Miller and Terry Greenlee (formerly of the Pennisula QL Group) made a surprise visit to the show.

The key visitor to the show was the renowned author Stan Kelly-Bootle, writer of many computer books, including the "Devil's DP Dictionary", and the holder of the first post-graduate degree in Computer Science (from Oxford).

Simon Goodwin has been a reader of Stan's work for a number of years (so have I) and considers him one of the few that have really mastered the English language. When I mentioned to Simon that Stan was coming to the Show, Simon's jaw literally dropped. Simon got to spend a few hours chatting with Stan, smiling the whole time.

There was no organized demonstrations or talks, just a whole lot of Sinclair talk going on. Since I was the organizer, I was not really relaxed enough to truly enjoy the show.

When the show ended at 5:00, there was a debate on where to have dinner held. Originally, Tony and Roy were leaving for the Airport right after the show. Unknown to me, their flight was delayed 4 hours and they had time for dinner after the show. Luckily everyone agreed on a place for dinner and the restaurant easily handled a group of 13.

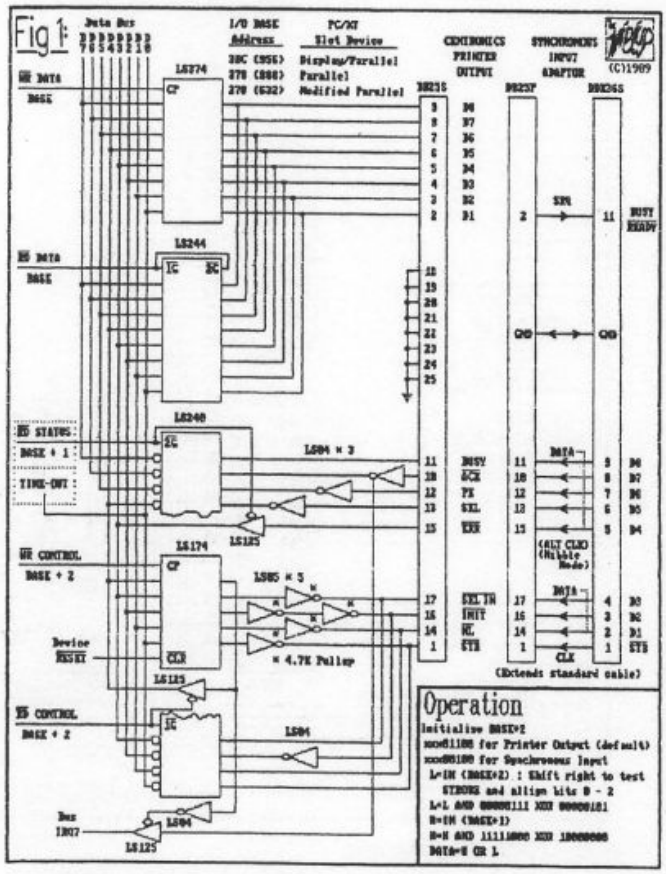
We are talking about plans for next year. We have just discovered a local "Vintage Computer Show" that would be the right audience to advertise our show. Hopefully we can generate more local attendees next year.

Improved TS-2068 Internal ROM Bypass I

by the late William Pedersen WIDJUP

A major barrier to fixing the problems in INTERNAL ROM is one of access. Even though the ROM is socketed, you still have to open the case. This risks damage; and even then, replacing ROM with EPROM requires wiring modifications. This article shows how to replace HOME ROM and EXROM without touching a screw.

It is not necessary to replace ROM with EPROM. Battery backed up static RAM will do as well if equipped with a READ ONLY switch. That way the RAM can be written and then switched over to write-protect state. The saving in time over burning EPROM is impressive, however, you will probably elect to use EPROM once corrections have been made.



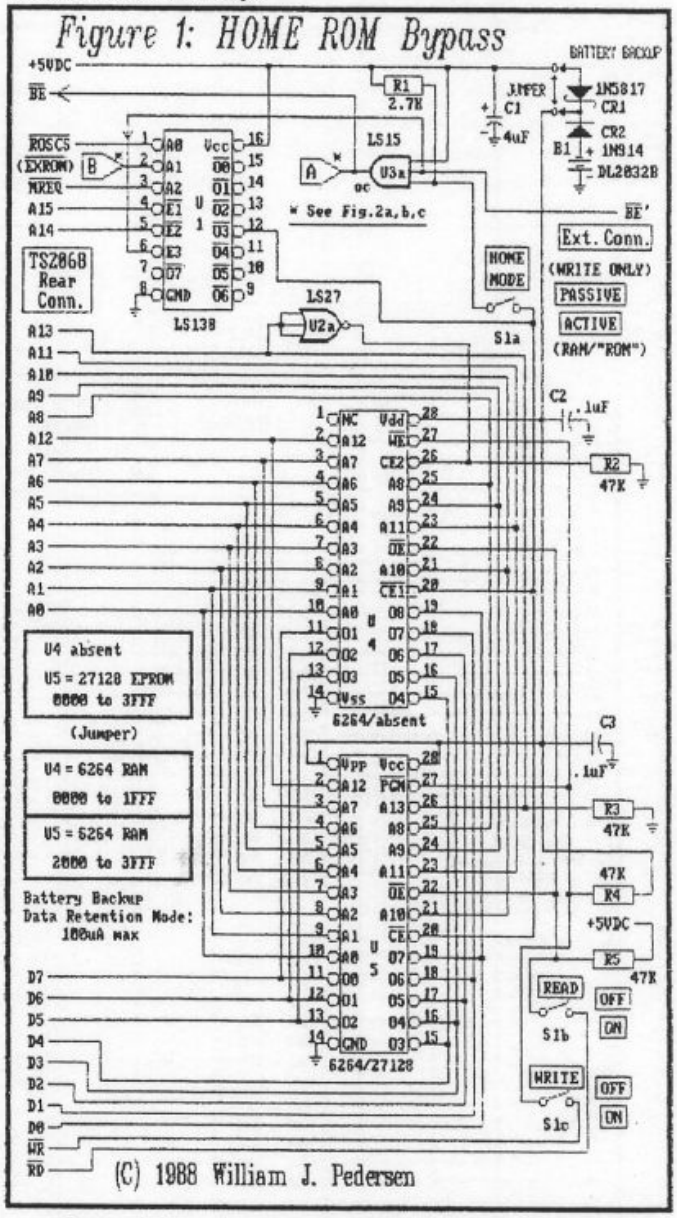
When expansion banks are connected, they assume priority by driving the BE (Bus Enable) signal low (using open collector gates). This is applied at the rear connector and sensed by the SCLD, which then shuts down all LOCAL memory (except for display memory while it is using it). It includes ROSCS, EXROM and ROMCS signals which enable DOCK (cartridge), EXROM and HOME ROM (memory chips) respectively. DOCK and EXROM are not true expansion banks in the sense that they have lower priority than external BANK #0 and #254. The "missing" three banks in TIMEX documentation are not really missing at all. The much-touted BEU would have contained a new operating system using them.

The design presented here has very carefully retained the ability to attach a non-TIMEX BEU with expansion slots -- or any other system your heart might desire.

Many users will not initially have a back-plane with expansion slots. The preferred location for ROM bypass would seem to be the cartridge slot; however, locating it there disclosed a bug which involved BE signal conflict on rare occasions.

Home ROM Bypass

Home ROM is enabled by the SCLD using a signal named ROMCS. Unlike EXROM and ROSCS, this signal is not available externally.



Our task is to disable Home ROM, and then enable an external equivalent.

The first is easy. Any time HOME ROM is being addressed, bus signal BE can be driven low. This suppresses ROMCS.

The second is a bit more complicated, but not difficult. HOME ROM is addressed whenever NOTHING ELSE is! It is a process of elimination.

Home ROM is **not** being addressed when:

1. A14 or A15 is high. (CHUNKS 2 thru 7 addressed),
2. EXROM is low (EXROM addressed),

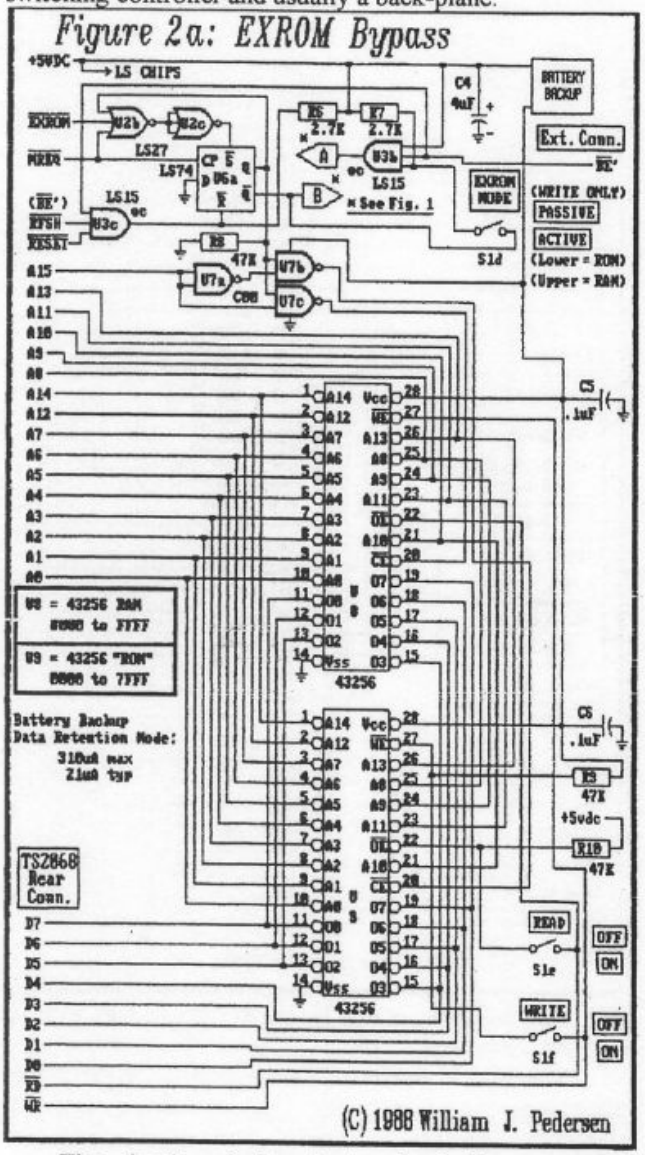
3. ROSCS is low (DOCK bank addressed),
4. MREQ is high (NOP, I/O or INTERRUPT CYCLE),
5. BE is low (Expansion bank active),
6. RD is high (Write cycle in process) or
7. RPSH is low (Applies to some static RAM).

What logic could be easier to implement?

Before attempting operating system revisions, a copy of existing code can be used in the bypass. This is an excellent test of bypass circuitry. No change in operation should be detectable when a bypass card is present, or absent.

EXROM Replacement

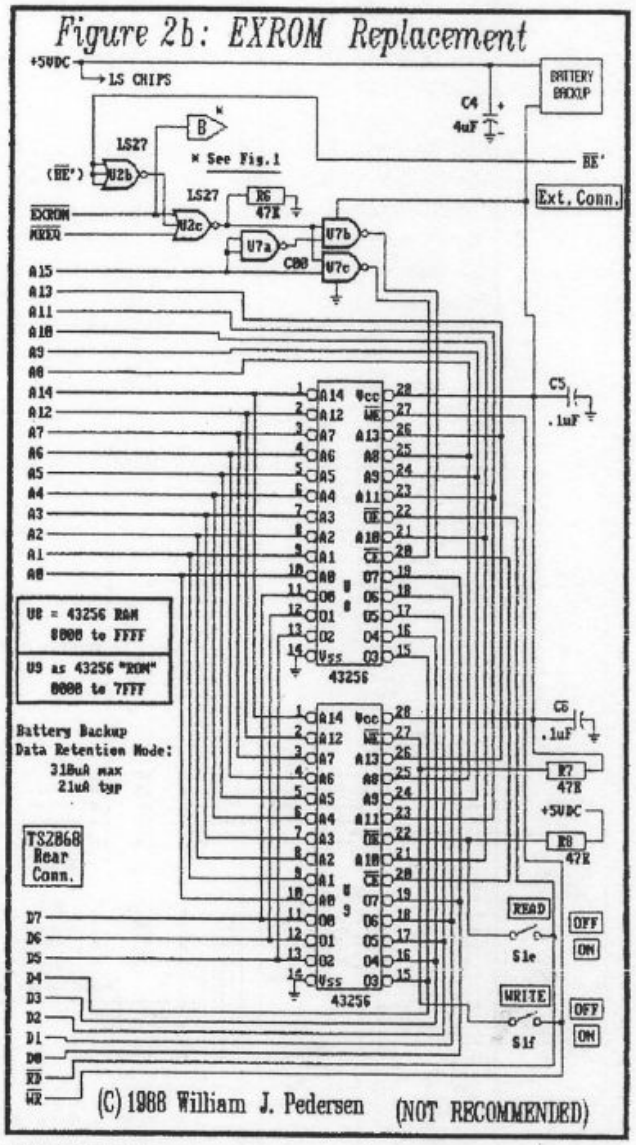
EXROM presents an interesting problem. It is enabled by the EXROM signal (not BANK #254), but lack of complete internal address decoding results in false images occurring in all other chunks, not just CHUNK #0. If we had an expansion BANK #254, it would have higher priority than the internal chip -- but that requires a bank switching controller and usually a back-plane.



The simple solution is to physically remove the EXROM chip from inside and remount it with full address decoding on a board. Though it is necessary to open the case, no wiring changes are needed and the chip can always be put back.

A tremendous advantage is gained when this is done. The false images are gone, making seven CHUNKS of EXROM available for use as either RAM or EPROM as desired. Again, this can be tested for no change in operation before and after the change.

Figure 2b is a practical variant of this which allows updating.



EXROM Bypass

Lacking a bank switching controller, it would seem impossible to use the BE signal to disable the internal EXROM chip and still have an external bypass take over. The presence of the EXROM signal from the SCLD would try to cancel itself, resulting in oscillation.

If the initial presence of EXROM signal can be latched on the bypass board, and then BE is switched low, this can be avoided.

Now we have a way to leave those screws untouched.

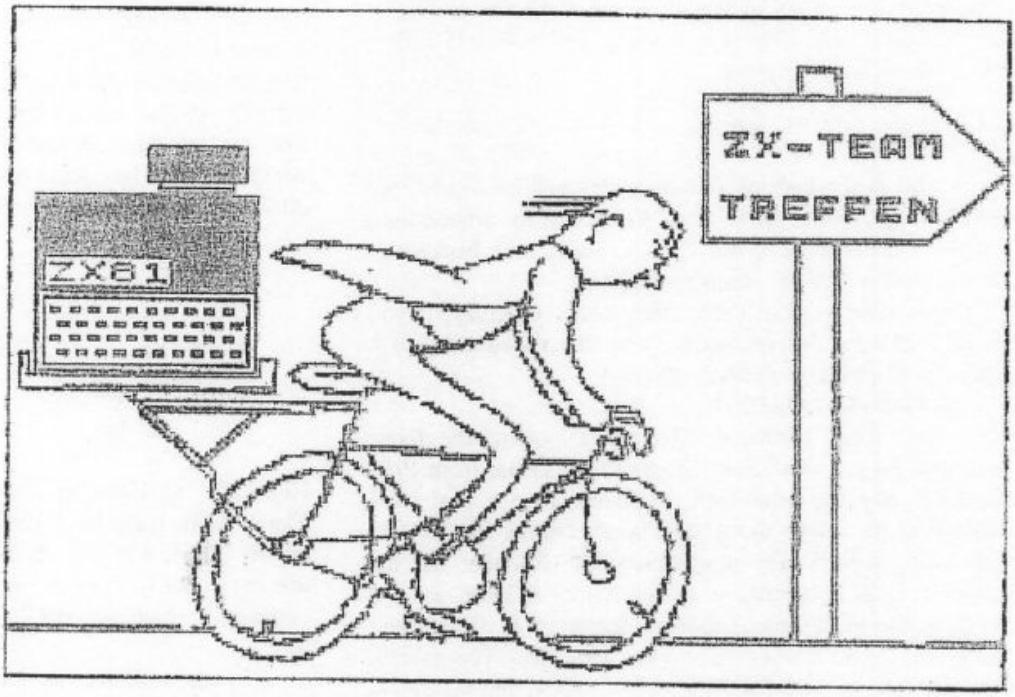
We need to reset the latch before the next instruction comes along, when an external bank exerts priority, at power-up and REFRESH.

The extra cost of a safe approach might make the risk in removing the EXROM chip seem worth while. That is what Figure 2b is for.

This is **not recommended** unless you are a confirmed hacker!

Dieter Thomas Herbert Stefan Joachim Norbert Gerhard Gotthard Peter Georg Ludwig Uwe Henning Michael Bodo Eckhard Gernot Thomas
Philip Kai Nickel Gerd Mike

ZXITEAM



For our english readers:

The third ZX-TEAM-meeting 26-28 march in Dietges was a great success:
we had 32 participants, 21 of them stayed over night! In our "exibition" a lot of ZX projects could be demonstrated. All new and a big lot of some elder of our developements could be seen in action.
Hard-disk controller with MEFISDOS, LCD-screen, hand-scanner, ZX-CAD and much much more. The ZX81-mailbox demonstration from Gernot was a great success!
New projects have been discussed like a SMD-ZX or the Jupiter-Ace project.
On the fleamarket everyone could find some good bargains!
For more information and some photos please look at ZX-TEAM-homepage, meetings-pages: `http://home.t-online.de/home/p.liebert/zx-team.htm`
If you would have liked to participate, please do not be too sad: the next great ZX-TEAM-meeting will be in spring 2000. You are welcome!!!!!!!!!!!!!!!

peter@zx81.de

How to Hack on The 2K Spectrum

Les Cottrell

Part 6 – cont.

```
CC94 3E C3     LD A,#C3
CC96 32 32 5B      LD (#5B32),A
CC99 21 A1 CC      LD HL,CCA1
CC9B 22 33 5B      LD (#5B33),HL
CC9E C3 00 5B      JP #5B00
```

This puts the instruction JP #CCA1 at #5B32 so the loader decrypter will return to our hack at #CCA1 when finished.

```
CCA1 21 B2 CC  LD HL,#CCB2
CCA4 11 D5 5B  LD DE,#5BD5
CCA7 01 08 00      LD BC,#0008
CCAA ED B0     LDIR
```

This copies the final part of our hacking routine to #5BD5, where it will be executed once the whole game has been loaded.

```
CCAC 21 00 00      LD HL,#0000
CCAF C3 35 5B  JP #5B35
```

The LD HL,#0000 instruction is important, because it's the instruction we overwrote with out JP back to the hack. Therefore, we've got to execute it, otherwise the loading system may crash. Then it resumes loading at #5B35 with the POKEs firmly in place.

```
CCB2 3E B6     LD A,#B6
CCB4 32 5F AB  LD (#AB5F),A
CCB7 C3 BC F5  JP #F5BC
```

This is the hacking routine which will be copied into the loading system. AB5F,B6 is the POKE for infinite lives (which can be worked out by a forwards or a backwards trace), and JP #F5BC jumps to the game.

And that's about it for Bleepload! Hopefully, if you were hacking a different game, you still managed to do it (they're all virtually identical anyway).

ULTIMATE LOADER

Remember Ultimate? They were one of the finest software houses of all time. Most of their games from 1983 to 1987 had the same type of loader (but a few were Speedlocked - more about them later). On the face of it, it just looks like a totally unprotected BASIC loader, but the appearance is deceptive. The five blocks it loads are the loading screen, the game itself, a decrypter at #5B80, and two very short blocks of system variables. The system variables are, in actual fact the BASIC clock, and determine how many 50ths of a second the computer has been switched on for. The decrypter works using this system variable. The upshot of all this is that if you stop the program for even 1/50th of a second, you'll mess up the decrypter. You can get round this with a Multiface by loading in the first three blocks of code, then replacing the code at #5B80 with #F3,#18 and #FE. This disables interrupts, so the system variable doesn't get updated, and causes an endless loop. Load in the last two blocks, activate the Multiface, and find out what the system variable should be. Then you can put this into the decrypter automatically.

MIKRO-GEN LOADER

This loading system appeared on just about every game released by the software house Mikro-Gen (oddly enough!) from about mid-84 to their demise in 1987. They come in two varieties, and you'll need a Multiface to hack some of the later ones, unfortunately.

The first type are recognized by black and white loading stripes, which loads in a screen block, and then the main game block separately. I'll be doing Pyjamarama as an example, but any Mikro Gen game which fits the above description will do.

So the first thing to do is to *Hack the BASIC loader.

```
    PYJAMARAMA LINE 0 LEN 504
    0 BORDER 7:PAPER 7:INK 0:BRIGHT
0:FLASH 0:CLS:PRINT AT
    12,12;"LOADING":RANDOMIZE USR (PEEK
23627+256*PEEK 23628+6)
    20 POKE 23756,0:POKE 23757,0:SAVE
"PYJAMARAMA" LINE 0:RANDOMIZE
    USR 33040
```

The BASIC loader actually features much more than what we can list. If you're old enough to remember the ZX-81, you'll recall that the best place to put a machine code program is in a REM statement. And that's almost the case here, except the machine code comes after the ASCII code #0D (NEWLINE), so you can't list it. But it's there. It's activated by the RANDOMIZE USR command. Type PRINT (PEEK 23627+256*PEEK 23628+6) and you'll find out the start address of the code. I made it 23984, which is 5DB0 hex (but you might find it to be something different), so disassemble this address.

```
5DB0 F3    DI
5DB1 31 00 00      LD SP,#0000
5DB4 2A 4B 5C  LD HL,(#5C4B)
5DB7 11 1C 00      LD DE,#001C
5DBA 19        ADD HL,DE
5DBB 11 16 80      LD DE,8016
5DBE 01 E7 00      LD BC,00E7
5DC1 ED B0     LDIR
5DC3 C3 16 80      JP 8016
```

Hopefully the DI and the LD SP,#0000 should be familiar. The next line loads HL with the two byte value starting at 5C4B. I made it 5DAA. This then has 1C added onto it, making it 5DC6. The rest of the code is a simple LDIR command, which puts the loading system to where it should be.

In our hack, we can simply use a headerless loader to load the code into place. We know that 5DC6 goes to 8016. BASIC always starts at the value in #5C53, which is #5CCB in this case. We know that the length is 504, or #1F8 hex bytes long, and the start address is (#5CCB-#5DC6)+8016 = #7F1B. So, run the following routine.

```
5B00 DD 21 1B 7F   LD IX,#7F1B
5B04 11 F8 01      LD DE,#01F8
5B07 3E FF     LD A,#FF
5B09 37    SCF
5B0A CD 56 05      CALL #0556
5B0D 30 F1     JR NC,#5B00
5B0F C9    RET
```

I've put a JR NC,#5B00 in, so that the computer ignores the BASIC header, and will only return on loading the main BASIC block. You should also note, that in the final hack, we'll have to add a DI and a LD SP,#0000 sometime. For now, disassemble #8016

```
8016 DD 21 00 40   LD IX,#4000
```

```
801A 11 01 1B      LD DE,#1B01
801D CD 4F 80      CALL #804F
```

This code activates the turboloader, which loads in the title screen.

```
8020 21 00 40      LD HL,#4000
8023 01 00 1B      LD BC,#1B00
8026 CD 3F 80      CALL #803F
```

This code verifies that the screen has loaded in properly (the routine at #803F adds up all the memory with start HL and length BC, and compares it with the byte after this block), and resets the computer if it hasn't.

```
8029 DD 21 00 82   LD IX,#8200
802D 11 A0 7A      LD DE,#7AA0
8030 CD 4F 80      CALL #804F
8033 21 00 82      LD HL,#8200
8036 01 9F 7A      LD BC,#7A9F
8039 CD 3F 80      CALL #803F
```

This is exactly the same as with the previous code, except it loads and checks the main game instead of the loading screen.

```
803C C3 89 FC      JP #FC89
```

Put a breakpoint over this instruction. Now POKE #8012 with F3, #8013 with #31, #8014 with #00 and #8015 with #00 (because we didn't execute the DI:LD SP,#0000 from the BASIC loader, and the game will not load otherwise), JP #8012 and start the tape. When the main game's loaded, disassemble #FC89.

```
FC89 21 EF B4      LD HL,#B4EF
FC8C 11 00 40      LD DE,#4000
FC8F 01 00 1B      LD BC,#1B00
FC92 1A      LD A,(DE)
FC93 AE      XOR (HL)
FC94 77      LD (HL),A
FC95 23      INC HL
FC96 13      INC DE
FC97 0B      DEC BC
FC98 78      LD A,B
FC99 B1      OR C
FC9A 20 F6         JR NZ,#FC92
FC9C C3 EA BE  JP #BEEA
```

This decrypter uses values in the screen memory, so you'll have to put a breakpoint at FC9C, put a JP #FC89 at #8029, JP to #8012 and reload the loading screen before you can run it. Then disassemble #BEEA.

```
BEEA 31 00 00      LD SP,#0000
BEED CD CC BE  CALL #BECC
BEF0 C3 00 82      JP #8200
```

This code puts the stack pointer back at #0000, CALLs another decrypter, and JPs to #8200, which is the start of the game. Change the #8200 to a suitable place to put POKEs; finish them with a JP #8200 to start the game. Here's the final hack, and I've put it at #5B00, because it doesn't get overloaded, apart from the byte at #5B00 itself, which is no longer needed by that time. Also, I've executed the DI:LD SP,#0000 directly, as well as the code from BEEA to BEF2.

```
5B00 DD 21 1B 7F   LD IX,#7F1B
5B04 11 F8 01      LD DE,#01F8
5B07 3E FF      LD A,#FF
5B09 37      SCF
5B0A CD 56 05      CALL #0556
5B0D 30 F1      JR NC,#5B00
5B0F 21 1C 5B      LD HL,#5B1C
5B12 22 3D 80      LD (#803D),HL
5B15 F3      DI
5B16 31 00 00      LD SP,#0000
5B19 C3 16 80      JP #8016
5B1C 21 25 5B      LD HL,#5B25
5B1F 22 9D FC      LD (#FC9D),HL
5B22 C3 89 FC      JP #FC89
5B25 31 00 00      LD SP,#0000
5B28 CD CC BE  CALL #BECC
5B2B AF      XOR A
5B2C 32 ?? ??      LD (????),A
5B2F C3 00 82      JP #8200
```

The other type of Mikro Gen loader is almost identical, except the whole game loads in one long block. Then end of the BASIC loading system is missing to start with, and is only loaded right at the end of the main headerless block. You can find out the missing code by loading the game as normal, then stopping it with a Multiface in the pause between the game loading, and the game starting (approx. 3 seconds), and hack it in the same way as Pyjamarama.

POWERLOAD

This protection system appeared first around the start of 1984, and was written by "Tag" (Phil Taglione) for Incentive Software. However, it's been used by quite a lot of other software companies as well, including Beyond, Mirrorsoft, Prism and Ariolasoft. It can be recognized by the screen turning black, accompanied by a few ascending beeps. It then loads one short headerless block, and then a longer headerless block, which includes the attribute file for the game coming up "backwards" i.e.: right to left, starting from the bottom. The game also stops loading just before the end of the long headerless block.

The only thing I know of that YS have put on the covertape that has Powerload is the Graphic Adventure Creator, but that's pointless hacking, so instead I'll be hacking Dynamite Dan. Of course, most other Powerload games are identical apart from some addresses, and, in fact, the BASIC loaders are all identical.

Before we start, I need to explain a little more about the stack, because Powerload uses it a lot. There are four commands which use the stack, and they are:

PUSH X (where X is any register) this takes the value in a register, and puts it onto the stack. The stack pointer then decreases by two (to be in the right place to store another value).

POP X

this takes the two byte value at the stack pointer (i.e.: the top of the stack), and puts them in a register. This also increases the stack pointer by two.

CALL XXXX

when you CALL a subroutine, the return address (i.e.: the address after the call) is PUSHed onto the stack, and the subroutine is JPed to. The stack pointer also decreases by two.

RET

when a RET instruction occurs, the computer takes the value on the top of the stack, and JPs to it. The stack pointer increases by two.

Now we've cleared that up, let's start hacking. *Hack the BASIC as usual.

```
D.D. LINE 0 LEN 496
0 REM
```

```
    10 CLEAR 59999:POKE 23693,0:POKE
23624,0:POKE 23697,0:CLS:POKE
    23659,0:FOR N=30 TO 36:BEEP
.075,N:NEXT N:RANDOMIZE USR
    24146:RANDOMIZE USR 0
    100 REM
```

The POKEs in line 10 just make the screen black and prevent you from pressing break. 24146 is #5E52 hex; but a breakpoint at #5E52 and GOTO 0. This is because the stack is set up in a specific way by the BASIC commands.

```
    5E52 F3     DI
    5E53 21 00 00      LD HL,#0000
    5E56 39     ADD HL,SP
    5E57 22 F2 5D      LD (#5DF2),HL
```

This code simply puts the value of the stack pointer into address #5DF2, so it can be retreived later.

```
    5E5A 31 95 5E      LD SP,#5E95
    5E5D 26 5E      LD H,#5E
    5E5F E5     PUSH HL
    5E60 21 68 5E      LD HL,#5E68
    5E63 E9     JP (HL)
    5E68 3E 12      LD A,#12
    5E6A 32 93 53      LD (#5E93),A
    5E6D E1     POP HL
    5E6E E5     PUSH HL
    5E6F D1     POP DE
    5E70 C9     RET
```

Put a breakpoint at 5E70 and JP to 5E52. At 5E70, the value on the top of the stack is #5E76, so a RET will JP to there.

```
    5E76 C1     POP BC
    5E77 7E     LD A,(HL)
    5E78 ED 44      NEG
    5E7A 77     LD (HL),A
    5E7B 23     INC HL
    5E7C 10 F9      DJNZ #5E77
```

This code is, as you might realise, a decrypter. The start value of HL is #5E12, and the initial value of B is #3A. In case you're interested, the NEG instruction turns the value in the A register into its negative form; in other words, the value in A is subtracted from #100 hex. Put a breakpoint at #5E7E and JP #5E70 which is where we left off).

```
    5E7E E1     POP HL
    5E7F 22 78 5E      LD (#5E78),HL
    5E82 C1     POP BC
    5E83 3E C9      LD A,#C9
    5E85 32 7E 5E      LD (#5E7E),A
    5E88 3E 00      LD A,#00
    5E8A 32 7A 5E      LD (#5E7A),A
    5E8D 5D     PUSH DE
    5E8E E1     POP HL
    5E8F C9     RET
```

This code changes the previous decrypter slightly, and RETs to 5E77. Put a breakpoint at 5E8F and JP #5E7E.

```
    5E77 7E     LD A,(HL)
    5E78 ED 67      RRD
    5E7A 00     NOP
    5E7B 23     INC HL
    5E7C 10 F9      DJNZ,#5E77
    5E7E C9     RET
```

This code works with the same values as the previous one; HL=5E12 and B=3A. It then RETs to 5E12. Put a breakpoint at #5E7E, and JP #5E8F (where we left off last time).

```
    5E12 21 B4 5F      LD HL,#5FB4
    5E15 11 B5 5F      LD DE,#5FB5
    5E18 01 B8 88      LD BC,#88B8
    5E1B ED B0      LDIR
    5E1D E1     POP HL
    5E1E 54     LD D,H
    5E1F 5D     LD E,L
    5E20 1C     INC E
    5E21 C1     POP BC
    5E22 ED B0      LDIR
```

These two LDIR commands wipe all the memory that isn't being used by the loading system. To get round this, you should change #5E1B, #5E1C, #5E22 and #5E23 to #00, to stop them being executed. Put a breakpoint at #5E24 and JP #5E7E (where we left off).

```
    5E24 06 1E      LD B,#1E
    5E26 E1     POP HL
    5E27 7E     LD A,(HL)
    5E28 EE A3      XOR #A3
    5E2A 77     LD (HL),A
    5E2B 23     NC HL
    5E2C 10 F9      DJNZ,5E27
```

The value in HL for this decrypter is #5E2E, which is right after the decrypter. To crack it, therefore, move the code from #5E24 to #5E2D somewhere safe (such as #5B00), put a breakpoint on the end, and run the code from there. When that's done, put a breakpoint at #5E2E and JP to #5E2E (so that you're back in the right place in the loading system).

Carrying on with the loader....

```
    5E2E E1     POP HL
    5E2F 22 02 5E      LD (#5E02),HL
    5E32 E1     POP HL
    5E33 22 05 5E      LD (#5E05),HL
    5E36 37     SCF
    5E37 3E 07      LD A,#07
    5E39 CD 00 5E      CALL #5E00
```

This code takes some values off the stack, and puts them into the subroutine at #5E00, which is then CALLed. Put a breakpoint at #5E39 and JP to #5E2E.

```
    5E00 DD 21 40 9C  LD IX,#9C40
    5E04 11 90 1      LD DE,#190
    5E07 14       INC D
    5E08 08       EX AF,AF'
    5E09 15       DEC D
    5E0A 3E 0F        LD A,#0F
    5E0C DB FE        OUT (#FE),A
    5E0E CD 62 05     CALL #0562
    5E11 C9       RET
```

This routine is a headerless loader. The start is #9C40 and the length is #190. Also the value of A is 7, and the carry flag has been set. In effect, we could have used a standard CALL #0556 headerless loader. Put a breakpoint at #5E3C and JP to #5E39. Start the tape and load in the first short headerless block. Then continue disassembling.

```
    5E3C D2 01 00     JP NC,#0001
```

This code resets the computer if there was a loading error from the first headerless block.

```
    5E3F 21 40 9C     LD HL,#9C40
    5E42 06 FF        LD B,#FF
    5E44 CD 77 5E     CALL #5E77
    5E47 06 FF        LD B,#FF
    5E49 CD 77 5E     CALL #5E77   (to be continued)
```

Unclassified Ads

Place your ads here, it is free!

Mail to: A. KAHALE 3343 S FLAT ROCK CT SIERRA VISTA AZ 85650-6874

Please inform and/or update the Editor of any changes in your ad/s

SPECTRUM for your 2068

If you are a LarKen LK-DOS owner and would like to run SPECTRUM programs on your system, we will supply a V2 EPROM, socket and 74HCT32 for $12 which includes shipping and handling. The installation instructions are in your LarKen manual. We shall not be responsible for your install job. AERCO owners need only the EPROM for $10 forwarded to LarKen.

Bob Swoger Address on page 2

747 Flight Simulator

So you like to fly, the 747 Flight Simulator for SPECTRUM by Derek Ashton of DACC. Requires a SPECTRUM equipped 2068. Supplied on LarKen SSDD or DSDD LarKen disk for $10 which goes to Derek now working at Motorola with Bob.

Bob Swoger Address on page 2

Home Electronics Service

We have been a part of the Sinclair scene since 1982, repairing **ZX Spectrums** for Sinclair Research in England.
We provide Sales, Service, and Software for the

QL, Spectrum, ZX-81 and Z88.

www.members.tripod.com/hes_computing/hes1.html
E-Mail 74601.1535@compuserve.com
Hours of Operation is Monday - Friday 1300 hrs. to 2100 hrs. central time zone.
Phone 210 661-4376
John R. Rish
Home Electronics Service
5222 Kazen Dr.
San Antonio TX 78219 USA

The John Oliger Co.

11601 Widbey Dr.
Cumberland IN 46229

The John Oliger Floppy Disk System
FOR THE TS-2068

Disk Works
EXPANSION BOARD
2068 User Cartridge
DISK BOARDS "A" & "B"
2068 Parallel Printer Port
2068 EPROM Programmer
2068/SPECTRUM Joystick Port
DFh Mapped Universal I/O Port board
User Manual only : $5.00 (Read before you buy)

Service For America's
Favorite Home Computers and Their Accessories

SINCLAIR

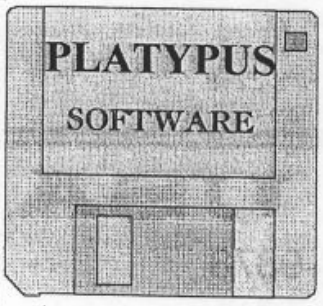
TIMEX ADAM ATARI IBM OSBORNE
TI COMMODORE TRS-80
BUY SELL TRADE UPGRADE

Repair Charge Examples

TS-1000, ZX-81, 1016 RAMPack, Memotech, ZEBRA Talker, MIRACLE Centronics, RAM Centronics.
$5.00 each + parts & shipping.
TS-2020, 2040, PC-8300, ZX-80, TI-99, Z-SIO, Byte-Back, AERCO 2068 Centronics, BASICare, LarKen RAMDisk
$10.00 each + parts & shipping.
TS-2068, Spectrum, A&J MicrD, Miracle 512K, LarKen 1000 & 2068 FDI, Kempston FDI, Cumana FDI, CST FDI.
$15.00 each + parts & shipping. July 1, 1996
Reasonable flat rate plus parts and shipping.
Write or call for prices, SASE appreciated

COMPUTER CLASSICS

RT 1, Box 117
Cabool MO 65689
Phone 417 469-4571 417 467-4571



QLAMBer	$20
QLuMSi	$20
SeekQL	$10
Upgrades	$5

Al Feng

914 Rio Vista Cir SW
Albuquerque NM 87105
(505) 843-8414

ZX-TEXT - Word Processor
ZX-CALC - Spread Sheet
Business Software
Cycle Accounting Financial Report Generator
ZX-CALENDAR - Time Management

ZX-81 TS-1000 TS-1500 TS-2068

Albert F. Rodriguez

A.F.R. Software®

1605 Pennsylvania Ave 204
Miami Beach FL 33139
305 531-6464

Make David an Offer
ZX-81/TS-1000 TS-2068
Hardware Kits
Real Time Clock I/O Controller RS-232
Centronics I/F 16K & 64K RAM 300 BAUD
Modem A-D Converter(assembled)

BYTE-BACK INC

536 Long Ter
Leesville SC 29070

WOOD & WIND

Bill Cable

ARCHIVE Based QL Software

QLerk - A complete financial program for the QL
- QLerk software (v3.21) with tutorial $29
- QLerk manual $29
- QLerk software & manual $50

DBEasy - A menu based database system
- DBEasy software (v1.6) $24
- DBEasy upgrade from V1.5 $7

DBProgs - A toolkit of ARCHIVE procedures
- DBProgs software (v1.8) $18
- DBProgs upgrade from V1.7 $7

DBTutor - A general purpose learning program
- DBTutor software(v1.5) $12

PC DBEasy - Just like QL DBEasy but, you must have PC ARCHIVE to use It.
- PC DBEasy software (v1.3) $12

Wood & Wind Computing
RR3 BOX 92
Cornish NH 03745 USA
Phone (603) 675-2218
Email : bcable@triton.coat.com

QBox USA

810 254-9878
24 hours a day
300 to 28,800 bps

Supporting all Sinclair and Timex users Fidonet echomail areas for Sinclair computers Lots of new files for you to download such asTS-2068 emulator for those who use a PC give us a call and let us know what you want to see
Message area and file area

QL International, Quanta, QL Hacker's Journal, Spectrum/2068, ZX-81/TS-1000, Z88, NetMail, emulators, pointer, FDFormat for QXL/QDOS, etc.

Sysop John J Impellizzeri
Co-Sysop Don Walterman
Utica, Michigan, USA

Domino Cubes

Z88

Hardware & Software
352 7th Ave. 15th Flr.
New York, NY 10001
Phone 212 631-7563
Fax 212 947-5069 Voice mail pager 917 490-8407
Domino.cubes@excelsior.net

LISTing Newsletter

The Long Island Sinclair/Timex Users Group

Robert Malloy, Treasurer
412 Pacific St.
Massapequa Park, NY 11762

NESQLUG NEWS

New England Sinclair QL Users Group

Ed Kingsley, Editor
16 Highland Avenue
Saugus MA 01906
(781) 233-3671 EdK4@aol.com

QL Hacker's Journal

Supporting All QL Programmers

Timothy Swenson, *Editor*
2455 Medallion Dr.
Union City, CA 94587-1914
swensontc@geocities.com
http://www.geocities.com/SilconValley/Pines/5865/

CATS Newsletter

The Capital Area T/S Users Group

Barry Washington
7044 Cindy Ln
Annandale VA 22003
301 589-7407
BBS 301 588-0579
Internet mf0002@epfl2.epflbalto.org

Nite-Times News

Newsletter

Chicago Area Timex Users Group

Robert Swoger
613 Parkside Cir
Streamwood IL 60107-1647
630 837-7957
CENG-108@email.mot.com

The Ramtop

Newsletter

The Greater Cleveland T-S User Group

Thomas Simon Editor
615 School Ave
Cuyahoga Falls OH 44221
E-Mail CIS 73177,333

Jon Kaczor Production
4568 Williamson Ave
Brooklyn OH 44144
75363.1127@compuserve.com

ZX-Team Magazin

Peter Liebert-Adelt
LUETZOW STR 3
D-38102 BRAUNSCHWEIG
GERMANY
Email: p.liebert@t-online.de
http://home.t-online.de/home/p.liebert/zx-team.htm
Amateur Radio: DK4BF@DB0FC.#NDS.DEU.EU

The ZX Spectrum 48/128 Emulator

for IBM & Compatables: Z80 Version 2.01

Turn your PC into a real ZX Spectrum 48/128!

⇒- Full Spectrum emulation, border, flash, beeper, Interface 1, Microdrive in cartridge file, RS232 input and output redirection to file, COM or LPT, joystick support, 128K sound through Soundblaster or internal speaker, built-in monitor,

⇒- Able to load ANY, even protected or speed-saved program from tape, to save to tape, to redirect tape loads and saves to disk for easy file access,

⇒- 2500 line English documentation.

⇒- Z80 processor amulabon including R register, inofficial instructions, inofficial flags,

⇒- Runs okay under DOS, Windows and DesqView,

⇒- Full source code of emulator and utilities included!

Runs on any 640K PC; too slow for practical use on PC/XT's but fast enough on AT's ; runs at about 100% on 16MHz AT's (can be slowed down on faster machines), uses VGA/EGA/CGA or Hercules.

This program costs US $20. You will receive a 3.5" DD disk (5.25" disks on request), and you'll be kept informed about updates. Please send bank notes (bills), name and address to:

Gerton Lunter
PO Box 2535
NL-9704 CM Groningen
Netherland

If you send a cheque, please add US $15 extra and allow 4 weeks for delivery.

Jochen Merz Software

The RomDisk! Fully functional!

See previous issue of QL Today. Extremely small board for the QL's ROM-Port, works like a permanent RAM disk, even if the power is taken away, contents are not lost!

The color drivers are coming. In order to be able to upgrade to the color drivers, you need to have SMSQ/E. The *normal* SMSQ which is shipped with QXL card will not be able to handle more colors. To benefit from the colors in the future, and benefit **now** from the extra features of SMSQ/E

SMSQ/E for the QXL

As Aurora owners will be able to use more colors when the color drivers are available, another offer for SMSQ/E for the Aurora plus GoldCard/Super/GoldCard:

SMSQ/E for the Super GoldCard

QL Games & Upgrades QL Applications
ProWesS + Applications

Jochen Merz Software
Im stillen Winkel 12
47169 Duisburg, Germany
☎ 0203-502011 Fax 0203-502012

Credit Cards accepted

http://www.j-m-s.com/smsq/
e-mail smsq@j-m-s.com

QL Today

QL Today is published by Jochen Merz Software. Jochen Merz has been supplying software for the QL for several years and has built up a good reputation for quality and fair trading. The representative in Britain is Miracle Systems Ltd. who take subscriptions and do the distribution.

Subscriptions

Germany (+German add-on) DM 70

England	DM 60	£25
Rest of the world	DM 70	£30

Back-issues are available for DM 12 (incl. postage)
Checks should be made payable Miracle Systems Ltd.

English Office

Miracle Systems Ltd.
20 Mow Barton
Yates, Bristol
United Kingdom BS17 5NF
Tel. +44 1454 883602 Fax. +44 1454 883602

Editor

Dilwyn Jones
41 Bro Emrys
Tal-Y-Bont, Bangor, Gwynedd
United Kingdom LL57 3YT
Tel. +44 1248 354023 Fax. +44 1248 354023

Items for the Timex\Sinclair Computer

Timeworks Programming kit #1 For T/S 1000 & ZX81 $4.95
Mindware Gulp Game Timex 1000 & Sinclair ZX81 $4.95
Timex Horace & The Spiders for the 2068. $5.95
Timex Sinclair 1000 software on tape
Chess (16K RAM) qty 5 price $2.95 ea
The Challenger I qty 17 price $2.95 ea
Mindware MW100 plain paper printer for the TS1000 or ZX81 computer made in the US. $45.00
All items are new. Please add $3.95 shipping to all orders.
MC, VISA, American Express. phone 717-748-1747
Keith Electronics
224 North Grove St
Lock Haven, PA. 17745

ZX-81 Kit

$30 Plus $10
for shipping and handling
TS-2040 Printers 16K RAM
We Do Not Ship Outside The US & Canada
Checks, Money Orders and Visa or Mastercard
Call 212-675-8414 **FAX** 212-675-8980
Zebra Systems, Inc.
122 W26th St. Suite. 904
New York, NY 10001
Shipping is by UPS ground.
Basics of Timex Sinclair 1500/1000 BASIC
A copy of this book is shipped free with each ZX-81 kit.
Allen Wolach, Separately priced $12.95
BASIC Basics for the Timex/Sinclair 1500/1000
A copy of this book is shipped free with each ZX-81 kit.
Michael Barnett//Simon Barnett, Separately priced $12.95
The Ins and Outs of the Timex TS-1000 & ZX-81
Don Thomasson, $12.95
Computer Interfacing Technique in Science TS-1500/1000
Paul Field & John Davies, $12.95
ZX-81 BASIC Programming (in Spanish language)
Steve Vickers (Traducido y adaptado) A copy of this book is shipped free for asking if you just pay for additional shipping cost. Separately priced $10

Sinclair Resources

Jack Dohany (Developer - 2068)
627 Vera Ave
Redwood City CA 94061

John McMichael (Developer - Graphics)
1710 Palmer Dr
Laramie WY 82070

Bill Russell (QL)
Russel Electronics
RR 1 Box 539
Center Hall PA 16828

Keith Watson (AERCO & Z80 Emulator)
41634 Amberly Dr.
Mt. Clemens, MI 48038

Rod Gowen (RMG)
14784 S Quail Grove Cir
Oregon City OR 97045-8843

Send them a LSASE and ask for information about their current products and/or services.

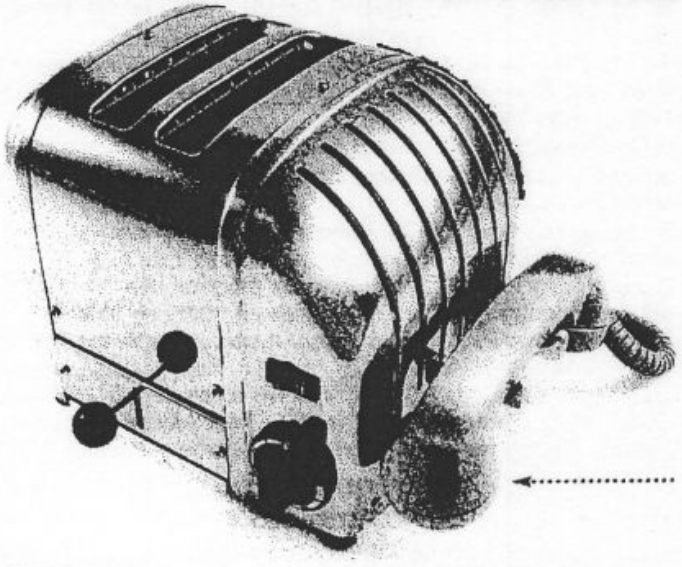
Alec Carswell

Motivation

16 Montgomery Ave
Beith Ayrshire KA15 1EL
United Kingdom

Surplus T/S Inventory

JOHN J SHEPARD III
281 130th ST
OGDEN IA 50212
< jshepard@netins.net >

JACK BOATWRIGHT
67325 FRYREAR RD
BEND OR 97701
< jboatno4@outlawnet.com >

QL TS-2068 ZX-81 Books Magazines Software