

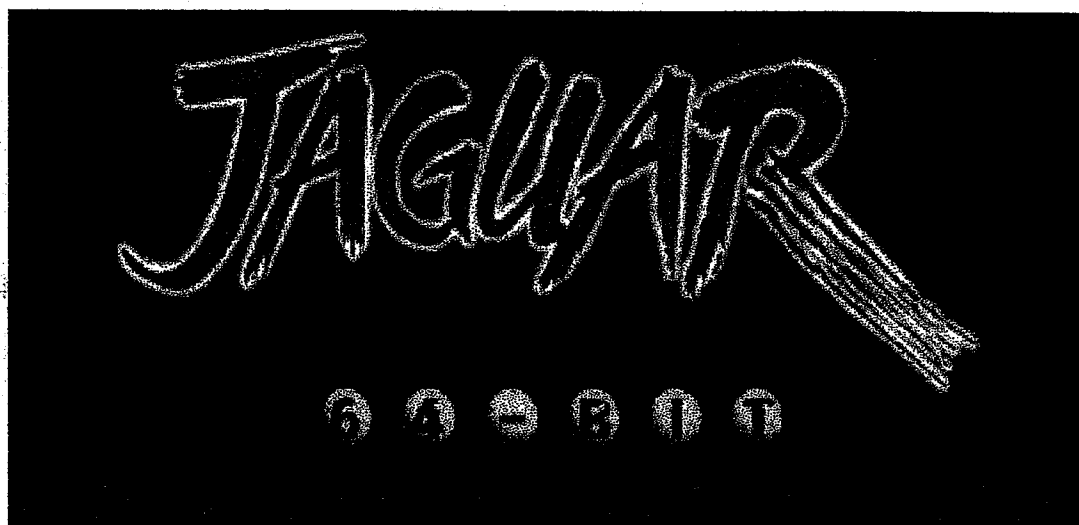
Issue: Summer 1996

\$2.50



NEW! GAZETTE

The Hamilton, Burlington, Oakville ATARI Users' Group Newsletter
for 8-Bit and ST Atari Systems



Win an Atari Jaguar 64! Get Your Ticket Today!

Inside:

HBO Happenings - Midi Notes - Internet Bits - Classified Ads
RAM/ROM Control on XL/XE - The Role of the HBOAUG

Executive for 1996

President

William M. Moeller 905-544-3387
af332@freenet.hamilton.on.ca

ST Vice President

Jeff Hostein 905-679-6943

8-Bit Vice President

Larry Popa 905-388-4784

Treasurer/Secretary

Randy Charlebois 905-545-1179

ST Librarian

Roger Perrie 905-578-9179

8-Bit Librarian

Randy Charlebois 905-545-1179

Correspondent

Glen Atwell 905-549-9078

Editor

William M. Moeller 905-544-3387

Members at Large

Vacant 905-555-5555

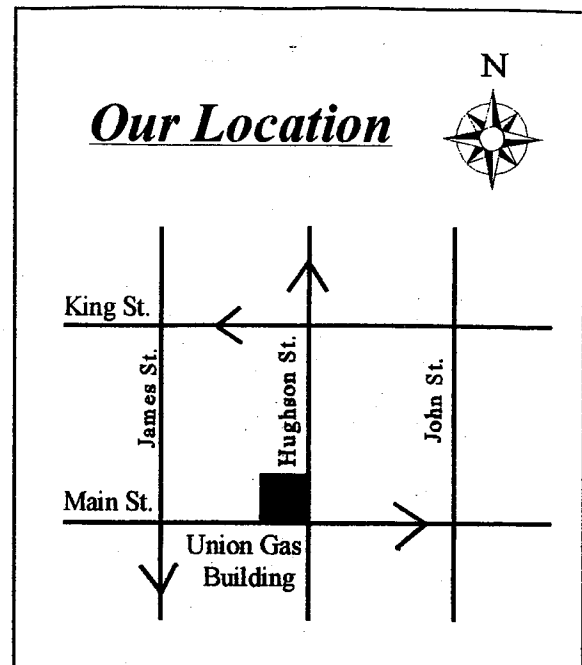
The H.B.O. Gazette is a publication of the Hamilton, Burlington, Oakville, Atari Users' Group P.O. Box 35543 StrathBarton Postal Outlet, Hamilton, Ontario L8H 7S6 E-mail address: ai065@freenet.hamilton.on.ca

Membership

Membership Dues are \$20 per person per year or \$3 per meeting. The first meeting attended is free! The executive reserve the right to waive this policy for specified meetings. Membership includes a Gazette subscription and enhanced access to the club BBS, H.B.O. Online at (905)573-3704 and preferred pricing of the club public domain disks.

Meetings

Meetings are held in the Blue Flame room in the Union Gas Building basement at 20 Hughson Street South in Hamilton at 7:30 on the Second Tuesday of each month from September to June. There are no summer meeting nights.



Contact People

General Information

Roger Perrie 905-578-9179

Atari 8-bit Programming

Larry Popa 905-388-4784

Atari 8-bit Hard Disks

William M. Moeller 905-544-3387

Atari 8-bit Telecommunications

William M. Moeller 905-544-3387

Atari ST Telecommunications

Glen Atwell 905-549-9078

Atari ST Midi/Music

Ron Polongio 905-318-5422

Atari ST Programming

Jeff Hostein 905-679-6943

Atari ST Hard Disks

Roger Perrie 905-578-9179

Atari Hardware Repair

Randy Charlebois 905-545-1179

Atari Jaguar 64

Larry Popa 905-388-4784

Message from the President

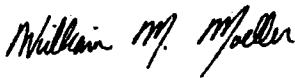
I was recently going through some old Antic and A.N.A.L.O.G. magazines, and came across an article entitled "8-bit Blues". You see, at the time, the Atari ST series had just been released, and the author was lamenting being stuck with obsolete equipment, unable to upgrade for cost reasons.

The final point of his article was that computer companies develop better computers at a quicker rate than average people can purchase them. Therefore, one must be happy with ones computer because there will always be a better more expensive one out there. It was true then, and it is true today!

Now, all Atari brand computers are obsolete. Should we jump ship, to the IBM or Macintosh world, just because our computers are no longer in popular use?

A resounding "NO" would be the response I would hope to hear!

If your computer is doing everything you want it to do, why change? Sometimes it is difficult to determine the difference between what you "want" and what you really "need". Many people not too long ago, laid out a lot of money to have the kind of system you have right now. I currently have a room full of surplus Atari equipment that was literally *GIVEN* to me! If I could transport myself back in time to 1982, I would be a very wealthy man selling it off for a fraction of what it was purchased for. As long as our machines are still working, they have depreciated to the lowest point they can. They are still very useful, entertaining devices. I would hope you would think twice before investing some \$3000 in new computer equipment. Wouldn't you rather keep your Atari, and have a vacation instead?



William M. Moeller



Table of Contents

Message from the President	2
H.B.O. Happenings	2
Internet Bits	3
Atari Video Gaming Scene	7
Treasurer Report	9
The Role Of The H.B.O.A.U.G.	9
8-Bit Librarian Report	10
ST Librarian Report	11
Midi Notes	12
RAM/ROM Control on XL/XE	14
Classified Ads	21

Next Meetings:
Tuesday June 11, 1996
Tuesday September 10, 1996

H.B.O. Happenings

By William M. Moeller

The H.B.O.A.U.G. is having a Jaguar raffle! Tickets are only \$5 and will be limited to only 50! That is pretty good odds. The system is brand new, and comes with the game Zool 2. The draw will be held at the end of the June 1996 meeting. See Larry Popa to buy your ticket.

As you probably noticed, the Gazette cover is in living colour! This is a first for our Users group. Speaking of the Gazette, back issues will be for sale for \$2.50 for each copy. You must pre-order. See Randy Charlebois to place your order.

The March 1996 meeting was very interesting. Ron, our resident Midi expert showed how he uses the Atari ST to compose and have fun with his music equipment. The small but appreciative audience was fascinated by this "one man band". Way to go Ron!

The April 1996 meeting consisted of a demonstration of the Jaguar games Space Ace and Rayman. The audience was impressed with the graphics of both these games. No doubt this helped sell some Jaguar raffle tickets! A tutorial was given on compression programs for the Atari 8 bit. This covered archiving, Discomming, and included a discussion of file transfer protocols. A parallel demonstration was conducted by the ST librarian to a decidedly small audience.

Our book librarian in the future will not be bringing in the entire book holdings of the club. Instead, he will rotate the books he brings. Regardless, if you have a specific request, feel free to ask, and he will be sure to have the book for you at the next meeting.

A reminder of the new policy of the HBO with regards to submitting an article for publication in the Gazette. If accepted, upon publication, the author will be entitled to a free disk of his or her choice from either club library. Just show the librarian your article.

The H.B.O. is purchasing your 720 k disks. Or, the club will exchange new 1.4 meg HD disks for your good condition used 720k floppies. See the ST librarian for details.

We need more members! If you bring in a new person who buys a membership, you will be entitled to an

incentive to be soon determined by the executive. These incentives could include discounts on your next year membership, club disks, or prizes. Just have the person identify you when purchasing a membership. Now, go out and find some Atari users!

**Call H.B.O. Online at
573-3704**

Internet Bits

Relevant Atari Information From Cyberspace

*From: keitai@usa.pipeline.com (Keita Iida)
Subject: Atari's 1995 4Q Earnings Report
Date: 17 Mar 1996 01:47:15 GMT*

Saturday March 16 7:31 AM EST

Atari Reports Fourth Quarter Loss

SUNNYVALE, Calif. (Reuter) - Atari Corp. is reporting a substantial loss for the last three months of 1995, reflecting a steep inventory write-down and poor sales of its Jaguar entertainment system.

The Sunnyvale, Calif.-based computer entertainment company said it lost \$27.7 million or 43 cents a share in the fourth quarter of 1995, compared with a profit of \$17.6 million or 30 cents a share a year ago.

Sales tumbled to \$2.8 million from \$14.9 million for the fourth quarter of 1994. The year ago period included a gain of \$29.8 million from a settlement of patent litigation.

Atari attributed the loss for the latest quarter to inventory write downs, software development costs and substantially lower sales for the Jaguar and related software.

Atari cut prices on the Jaguar 64-bit multimedia interactive entertainment system late last year to \$99.95 in an effort to boost sales but the company said sales continued to be poor in the first quarter.

In a statement, the company said it is currently testing

different price points and software bundles for the system to sell its inventory.

For the year, Atari lost \$49.6 million or 78 cents a share vs. a profit of \$9.3 million or 16 cents a share in 1994. Sales in 1995 were \$14.6 million compared with \$38.7 million in 1994.

On Feb. 13, Atari and JTS Corp. announced plans to merge. Under terms of the agreement, the new corporation will operate under the name of JTS Corp and the officers of JTS will become the officers of the merged company.

The Atari entertainment business and the JTS disk drive business will operate as separate divisions of the new merged company.

As a result of the deal, Atari stockholders will hold approximately 60 percent of the outstanding shares of the new company following the merger.

Copyright 1996 Reuters Limited. All rights reserved. Republication or dissemination of the contents of this screen are expressly prohibited without the written consent of Reuters Limited

Comments to: reuters-admin@yahoo.com

From: schallt@worms.fh-rpl.de (Torsten Schall)
Subject: Atari Trade show Schreiersgruen GER
Date: 25 Mar 1996 17:45:27 GMT

Hi to all out there! Schreiersgruen greets the whole world!

It's done the show is over the visitors are happy. Let's have a look back to SA. the 23 of March. Let's do the timewarp again..

The show opened at 10 o' clock in the morning. After an hour entered the people massive all Trader's and the air was melting at 12 o' clock. The show was held in a ca. 90m*m great room in the first floor of a pub. After a few hour's the place between the several Trader's was filled with Software and Hardware thirsty people. The show was melting after the people had gone down in the pub to get back power from a great meal and came back to the show. The people where refreshed and ready for the final buy and talk round. At 17 o' clock was the show closed, with a smile on Traders and buyer's face. There where following Trader's:

Ke-Soft with many Infocom Adventures, Seven City's of gold, Gemstone Worrier, Lords of conquest, XE game system Soft, Classic's and some new Polensoft. He was very happy because he was over run after an hour and it lasted many more.

There was also AMC Soft/Portronic and the interested people have been happy because they bought the multipad 3, Sound Blaster, Stereo Phaser, CX85 10er Tastatur, Herbert, Print Star, Super Ski 3D and many more of Armin's Hard-/Software stuff. They could also test their new stuff.

The place where Kaisersoft was, was loaded with great amount of Polensoft and classic' s. The visitors could play and buy the latest Polensoft about 10+ new games. The could watch the latest Demo's from Poland and other's like the RAM Demo's and buy other Soft from Kaisersoft.

There was an extra place for Highlander Soft with it's Turbo Basic Distribution, Ram Demo's and a new build Book Order Service. Raimund Altmayer was showing and introducing the ST-XFormer, including a manual he has written for easy "start and use" guide for the ST-Xformer. He was over the whole show very busy to answer the questions and show demo's and other stuff.

The Lengenfeld ABBUC Regional Group sold many hardware and Software. He was the main guy for the organization, by the way he did a perfect job, thank's to Helmut Weidner one reason more for the great show. He stated that he will make the organization of the show next year again.

A little place was reserved for the new game system of Atari. There was the Jag and the Lynx. Both showed many of their high rated Software games. The people could play for free and test how strong these systems are beyond the "OLDIE" Atari 8 Bit. The people enjoyed to rest for a while before they become back the Konsumrausch.

The many people who showed and sold Hard/Software have had great business. The Organization and Trader where very surprise of the great success of the show and will come to the next show in Schreiersgruen 97. It was a great surprise because the date for the show and the place have been told the people very late and many German Disk Mag's have not reported about the Show.

Ok Fact's for you to understand how great the success was: 70+ visitor's payed the symbolic 1 Mark, About 10 Traders in a 90m*M room. This is now the show that can compete with the ABBUC JHV the greatest Show in Germany for Atari 8 Bit Systems. The organization and Traders believe that the show has been visited by more people if they had known it earlier.

So if you are a German or live near Germany (England, France, Belgie, Italie, Austria, Swiss, Poland, Techslowakia) try to visit our next Atari 8 Bit shows. You won't be disappointed but always ask before for Soft/Hardware that is there. A very little people couldn't find what they searched.

WE Kaisersoft (Polensoft/Top Mag RAM Musik Composer) Highlander Soft (ST Xformer/Turbo Basic Distribution / Ram Demo Coder) Ke Soft (Atari Software/Musik Programmer trader of many games) Helmut Weidner (Organization ,Lengsfeld ARC) and AMS (Andreas Magenheimer Soft, Dos Freak, Software and Hardware Tester) me ST org (Internet German Atari 8 Bit Show/soft/hardware Distribution, St Xformer, Internet Downloader Porter to XL) and the other second hand trader are very pleased with the show and will support it next year.

Torsten Schall
ST org
schallt@worms.fh-rpl.de

Newsgroups: rec.games.video.atari
From: llamaman@ix.netcom.com (yaK)
Subject: YaK's Quick Intro to VLM Hacking
Date: Sun, 31 Mar 1996 07:06:40 GMT

Hacking the VLM - A Brief Introduction

1: The Good News

There is a backdoor left in the VLM which allows the user to get at the edit mode that was used to create the banks of VLM effects. You can get in there and roll-your-own FX, and it's not hard to come up with stuff that is a *lot* better than many of the default FX.

2: The Bad News

There is no provision for saving your hacks. They are entirely transient, and you can wave them bye-bye as soon as you switch to another FX bank - and

unfortunately, whenever you open the lid of the Toilet to change the CD, it switches banks into a special Jaguar-logo bank, and it's bye bye hacks. Bummer, I wish it didn't do that, 'coz otherwise I'd just leave my VLM on all the time and evolve 9 banks worth of top hacks...

Also, the edit interface is buggy and pretty user-hostile; it's got no error-checking and it's possible to crash the VLM, or slow it down to a crawl, if you tweak certain parameters out of range. Hell, it was never intended for end-users, only for use by a certain deranged bovine.

3: Why Bother Then?

Because it's fun. Because if you're interested, you can find out a lot about how the VLM works. Because a lot of the default FX are fairly nonoptimal, largely 'coz at the time I was designing the banks I had pneumonia and was feeling like shit, and I always find that I make better FX when I feel good. And although you can't save the hacks, you can do what I do, and keep a tape in the VCR and lay down some vid whenever you come up with a particularly nice one.

4: How do I get there?

Easy. Select the effect you want to base your hack on. Then, in VLM-mode, hold down *,1,3,0 on the Jaggi control pad. This should bring up a menu called Spectrum and Triggers. That's not it though - that was just a red herring to confuse people looking for the hack ;-). While on the Spectrum and Triggers screen, press up,down,up,down... 8 times (so the cursor wraps from Trigger 1 to Trigger 5 and back again 8x) and then the display will change to Edit Mode. At that point you're in.

5: What do I do when I get there then?

Okay, first it helps to have some idea of what is going on inside any particular VLM setting. Each individual setting consists of up to six effects, each of which can take a shitload of parameters, which define how that effect changes over time. Parameters can be attached to waveform generators, to user control from the joypad, or to the spectrum triggers to create an audio-responsive effect. The most common types of effect you will see in the VLM are:

Digital Video Feedback area: DVF is what gives you all

those cool swirly screen-filling patterns and persistence fields. It is, however, quite an expensive effect in terms of proc power (one is throwing rather a lot of pixels around doing scaled rotates of the entire screen)... if you can live with a smaller DVF area on some of the effects the speed and responsiveness of many of the effects can be greatly enhanced. I have a whole new class of 60Hz feedback FX that I could kick myself for not putting in the default banks...

Draw Spectrum As Intensities: This basically plots the output of the FFT through the symmetry generator. On its own, or combined with DVF, this is a good setting to use for effects that you want to be precisely audio-reactive. Put it over some nice DVF or a persistence field, llovely.

Draw a Ring of Pixels: does just that, through the sym generator, with various parameters applied so you can change phases, number of points, pixel size, etc... a very versatile thing, simple though it is.

Draw Plasma: draws a tile of plasma on the screen. Useful for putting in the middle of a nice DVF field as a source effect. Can be programmed to flash on trigger band events too. Try hooking oscillators up to the XY position when it's in a 60Hz DVF field - yummy.

Empty slots cannot be edited. Time was when pushing B on an empty slot would bring up two pages' worth of FX that you could build in there, but sadly that came out. I wish I had never taken it out now. Damn, and I wish I had not removed Save Bank either...

There are a few others, starfields and the like. Play with them. Some of them are quite cool.

Fiddling Around With Stuff

When you press B to edit an effect, you are presented with three choices - Edit source function, which allows you to see and tweak the parameters that make up the basis of the effect; Edit symmetry generator, which allows you to twiddle with the sym generator, for effects that go through that pipe (not all FX do - DVF and the plasma don't use the sym gen at all, so changing stuff on their sym generator pages does nothing useful); and Edit Source Waves. Each effect has 8 programmable waveform generators which can be attached to any of the parameters defining the effect or the symmetry. Edit

Source Waves is where you go to adjust the speed, waveform etc of the waveform generators.

Edit Source Function

If you enter this menu, you will be presented with a list of the variables which control how that effect is displayed. In a DVF effect's Source Function menu, for example, you will see parameters controlling the Window Size, Scale, Rotate Angle, etc. You are free to cursor around this menu, twiddle the values, and see what happens to the display. (It helps to have a CD playing while you fiddle, so that all the effects are active and you can see what you are doing).

Menu items with (X,Y) after them represent pairs of variables. When you select a variable to edit, you will be presented with a slider, a 2-way slider, a position marker, or cross hairs, depending on the effect. Just use the joypad to change the values. Anything you change will stay changed for the life of your hack (that is until you change banks or open the Toilet).

While editing a variable, you may see the message 'Press * to Attach Waveforms'. If you press *, it takes you to the Attach Waveforms screen, which has a totally crappy and confusing UI, but which is nonetheless the key to the cooler aspects of VLM usage. It works in conjunction with the Waveform Edit page, and in a sane and rational world, would have been integrated in with that page.

The wonderfully-informative Attach Waveforms screen presents two rows of the numbers 1-8. The top row represents waveform generators linked to this variable. If one or more of these numbers are highlighted, it means there are generators already linked to this variable.

You can attach or detach a generator to the variable by pressing the corresponding number key 1-8. You can attach as many generators as you like - the resultant waveform is the sum of all the input waveforms.

You also can adjust the amplitude of the waveform here. Dismally, there is no display actually showing you the amplitude - you just have to watch the display to see the result and kinda fish about. Also, for some inane reason, if the waveform is attached to an X-component you adjust the amplitude with left/right, and if it is a Y component you use up and down. I think I was planning

to put in an actual display of the waveforms here, but I got ill and never finished it off. As it is, it's sucky and counterintuitive. Oh well...

The bottom row of 8 numbers allows you to modulate the attached waveform(s) with the output of an envelope generated by any of the five spectrum trigger generators. You press #, then the number of the trigger 1-5 (6, 7 and 8 are for the three joypad buttons) to toggle the trigger attach. And for some reason, like it was never properly debugged, when you toggle a trigger off, often the digit in the bottom row will not de-highlight. Oops.

After you have attached waveforms, you will probably want to mosey on over to the Edit Source Waves menu. Here, you can select a waveform generator by cursoring up and down. While the cursor is on a generator, that wave is displayed in the blue box. You can increase or decrease the frequency with A and C, and slip the phase by Left/Right. You can change the wave type by pressing a number on the numeric pad. User X and User Y translate to the 'position' controlled by the joypad in VLM Interactive mode.

Edit Symmetry Generator is a very interesting place to be for all those particle effects and anything else that goes through the sym generator. In this menu you can twiddle, attach waveforms to and generally piss about with everything to do with symmetry.

The sym generator uses two basic kinds of symmetry, planar 8-way reflection, which is quick, and rotational, which is very nice but considerably slower. The two types can be combined. Many of the items on the Symmetry page only apply to Rotational symmetry, and will have no effect if Rotational is switched off.

You set the sym types under the Symmetry Type menu, not surprisingly. Pressing 9 on this page toggles rotational sym off and on, and due to quite possibly the same bug as on the Waveform Attach screen, often the '9' in the middle will be incorrectly highlighted. It's usually pretty obvious when rotational is on though. Keys 1-8 toggle on and off the planes of the 8-way symmetry. Set up your types, and then go and have a fiddle with the parameters as in the Edit Source Function mode.

Since you can't add or delete effects, the nature of your hacks will be in part defined by what effect types are in

the bank you choose to edit, so if you want to do a lot of DVF stuff, try bank 3 or 4, and if you like particles, bank 1 or 2, and so on. You can switch between, and hack, all effects within a bank, but *if you change banks they are lost*. Be careful!

And, basically, there we have it (as I said to Flossie as I led her into the concealing darkness of the sheep-shed). Enough stuff to get you started playing about with VLM editing. Apologies for the crappy UI, but I never had time to really bring it up to scratch before I got that bloody pneumonia. As it is it was about a year between finishing the VLM code and the release of the CD-ROM, and I wish I had been able to spend a few more months at it. Both the FX and the edit mode could have been a lot nicer! Oh well, next time... I've already got some awesome stuff running on [closes his muzzle and remembers the three letters N, D and A] :-)

and oh yeah, I better mention:

CAVEATS: Here Be Bugs'n'Beasties!

The error checking on the UI is at best sucky and at worst nonexistent. It is quite possible to either choke up the VLM with some incredibly intensive sym mode that will have it doing one frame per Sunday, and you can even kill it with an honest to Ghu, thank-you-and-goodnight, little-silicon-legs-in-the-air crash. There are a few danger areas which I shall warn you of now:

- Positioning a DVF window too far off the edge of the screen can kill the system

- Attaching waveforms to DVF window size and position can be fatal. Look out!

- Attaching waveforms to the Rotational Symmetry Order can cause it to wrap to negative, which translates to something greater than 32767, which, as a sym order, will cause the system to choke most heinously. Won't kill it, but the frame rate will be measured in minutes per frame hehe...

Don't worry, killing the system will not do any permanent damage. The default banks are tucked away snug in ROM and will be restored when you restart the system, no matter how badly you take it down.

Well, I shall go and post this now... have fun and Happy

Hacking!

(:-) - the Beastly Boanthrope

Atari Video Gaming Scene

By Larry Popa

Hey gamers! At this point the Jag has quite a few carts out for it and the CDS are catching up quickly. I recently picked up the CD games Space Ace and Hoverstrike: Unconquered Lands, and I will be reviewing both games here this month:

Hoverstrike: Unconquered Lands

First Impressions:

Hoverstrike is a cool cartridge game that has been out for almost a year for the Jag. Shortly after the release of the CD unit, Hoverstrike was revamped for the added capabilities that the CD unit could offer. The improvements include: Full Motion Video (FMV) animation sequences in between missions, CD quality music, ten extra missions, and better control of the hovercraft. All of the original levels and missions of the cartridge version of Hoverstrike are included. When I reviewed the Hoverstrike cart for the Summer '95 issue of the HBO Gazette I gave it 14/15. Let's see how the new version holds up.

Graphics: *****

The graphics of the game play itself are nearly identical to the original cart version of the game. The only significant difference in the graphics of the game play is in the night missions. In the original the graphics for the night missions were all done in shades of gray with no details on them. The CD version instead shows the normal colours and details as in the day missions - it instead just darkens them to a point where they're barely visible. The FMV sequences that were added are amazing to look at and are just as good as the movies in Blue Lightning.

Sound: ****

The sound effects are mostly the same as the original with the exception of the photon cannons of the

hovercraft - which is only a minor change. The music is almost the same as the original as well, except that it has been remixed to CD quality. Although there are many different songs they all sound a little too similar to each other, and a little creativity could have helped.

Game play: *****

The controls have been significantly changed for this version of the game - and the changes are for the better. A reverse button has been added to the game pad, and is much more desirable than the stupid Down, A, and C combination that was required before. The controls can also be configured for whether the craft drifts when it stops, whether it rocks when it is hit, whether it takes damages for falls, etc. The game play itself is the same as the original and that is good. The original was an excellent game of strategy and skill and that quality has been maintained in the CD version.

Overall: 14/15

There hasn't been enough changed in the CD update for me to warrant giving Unconquered Lands a better review than the original and as such the original 14/15 stands. It would have been nicer if Atari had composed completely different levels for Unconquered Lands and as such it could have been considered a sequel rather than just a mildly souped up re-release. The addition of ten new missions is not enough to warrant the purchase of the CD version of Hoverstrike to those who already own the cart. To those who own neither the CD version of Hoverstrike is well worth having and is one of the Jag's best games.

Space Ace

First Impressions:

In these days where there is a plethora of high-end game systems out there seems to be more and more fluffy eye candy games come out that offer nothing more than some FMV sequences to amaze the player (or viewer in this case) without any real game play to back it up. Space Ace is one such game. The graphics are astounding, and look just like a real cartoon. The sound is equally good. Unfortunately the game play is almost non-existent. The premise behind the game is that you are Space Ace - a hunky, blonde-haired, blue-eyed super hero who has been zapped by an infanto ray that turned him into a squeaky-voiced, dorky little boy who now must recover his masculinity and his girlfriend that has been kidnaped by the evil Borf.

Graphics: ****

Space Ace looks great, the animation is smooth and beautiful. Don Bluth (the artist behind the movie the Secret of Nimh) was responsible for the artwork in this game and did a great job. My only real complaint is that the animation is slightly grainy but I think that the graininess was probably necessary to hold so much animation on a single CD. Perhaps a two CD package would have been in order.

Sound: *****

No problems here. Excellent CD quality music with tons of quality speech, and sound effects. If it only played as well as it sounds - sigh.

Game play: *

The game play is wretched. At certain points in the animation you are expected to make your move, which entails one of five actions (the four directions and fire). This would be fine it - if you had the choice of several of these actions. But instead, at any given moment, there is one and only one correct movement. As such there is no real game play to speak of, and it is a very frustrating to play. Even in the trickier parts where the more obvious movements don't work it still isn't that difficult since there are only five different possibilities.

Overall: 10/15

In spite of the good graphics and sound this game was not worth the \$50 I spent on it. If you have the money to buy this game then I suggest that you buy a different game and enjoy it.

H.B.O. Treasurer Report

By Randy Charlebois Secretary/Treasurer

H.B.O. ATARI USERS GROUP FINANCIAL REPORT April 1996

For the month ending April 1996 our fiscal balance was \$81.00 with no outstanding debts. Our major expense this month was the Jaguar for the raffle and

photocopying. The Jaguar will be given away at the June meeting. Buy your ticket soon as only a limited number are being sold. Thanks to those who bought raffle tickets last meeting. A successful draw will assist in funding our club.

Meeting expenses include pop, donuts, raffle prizes and the newsletter. Our revenue comes from memberships, raffles/50 draws, and disk sales.

Help support the club by buying disks from the library and draw tickets. A volunteer to run the draws and raffles would be greatly appreciated.

Involvement by our members will allow us to support new and old users. Thanks to Eric Oldfield for donation of his 1040 and other ST equipment.

The Role Of The H.B.O.A.U.G. An Editorial

By The Gazette Editor

Ask any new member of the H.B.O.A.U.G. what they expect from their membership, and compare it to what actually happens at the meetings, and you will get two different answers.

A cursory look at this dilemma may lead someone to believe that the club is not fulfilling its mandate. Unfortunately, this is not the case. I say unfortunately because if it were otherwise, we could "fix" the problem and continue on our merry way.

I was recently told by a relatively new member, that the said member had, I quote; "*Not learned ONE thing!*". He further added, "*Obviously this was not the club for him!*". This statement no doubt, made me question my involvement with the H.B.O. executive. It is obvious to me, that many have misinterpreted what their \$20 yearly membership buys them. Let me state right here, that this is **NOT** "*Atari school*". The purpose of the H.B.O.A.U.G. as laid out in the constitution is; to enhance the knowledge of individual club members concerning Atari microcomputers while, at the same time, providing a forum for the useful exchange of information and ideas.

In my humble opinion, the key word here is "*exchange*". This word implies a give and take. Perhaps I am wrong,

but the impression I get, is many new members expect to be handed what they consider to be "useful information", without even having an inkling of what that may be. They expect to have an expert on their machine, spend one or two hours during the meeting "one on one". They expect to have this person make them an "expert" with just a few "pearls of wisdom", and get frustrated at what they consider vague answers to their questions.

This person who told me he had not learned one thing while at the H.B.O.A.U.G., was one of the members who did not bother coming to the group tutorial we ran that very evening! Some nerve!

The current executive sacrifices many hours of their leisure time to put together meetings that are useful, instructive, and most of all fun. The executive does not get paid to do this work. They do not benefit financially from helping others use their Atari computers better. They do it as a community service. The club membership are not CUSTOMERS, and we are not selling a service here. The attitude of the said member was that he was a dis-satisfied customer, and I was about to "lose a sale". Well, I say "see ya later"!

The members of the executive ALREADY know how to use their machines! The only one losing out here is him.

The H.B.O.A.U.G. is a resource. Use it! Buy disks, ask questions to other members, not JUST the executive. Introduce yourself to other members! Volunteer to do a demonstration. If you have questions, you can always ask at the club announcements section right at the beginning! While I am at it, I would like to state for the record that I will not tolerate people who refuse to sit in front of me while I do the announcements. In the future, I will not begin until I have the members attention and the courtesy of silence, except when asking questions or making comments. If you are not here to hear the club announcements, why did you come? And, finally, ask yourself if you are like the individual I described above. If so, what are you going to contribute to YOUR club?

Atari 8-Bit Librarian Report

By Randy Charlebois

Lately I've been asked how to run files with different extenders in their names. This month I'm going to cover

filename extenders and what they mean. I'll also explain how to run or read files.

Filenames are used to distinguish one file from another. They consist of eight characters of any combination of capital letters and numbers, but the first letter must be a capital. No blanks, special characters (such as \$, @, or #), or punctuation marks are allowed in ATARI filenames. e.g. FILENAME

Filenames can have a one to three character extension. Extenders can consist of any three characters or numbers. They are mostly used to indicate the file type. e.g. FILENAME.EXT

I'll cover some of the most common extensions with a brief description of how to use them.

Extension Description

SYS System files. Files which contain system programs like DOS or BASIC. These files load automatically on boot up.

BAS Files which contain BASIC programs in SAVE statement format. These are "LOAD"ed from BASIC.

LST Files which contain BASIC programs stored in LIST statement format. These are "ENTER"ed from BASIC.

ASM Assembly language programs in source (text) form. They are created by ASSEMBLER EDITORS and used by COMPILERS to create machine language programs. See your assembler manual for more details.

SRC Source code for assembly language program. Requires assembler to compile.

OBJ Object files. Assembly language programs assembled into machine language. These can be run from DOS under option L. BINARY LOAD.

COM See OBJ.

EXE Execute file. See OBJ.

BIN Binary file. See OBJ.

M65 MAC65 file. Created by the MAC65 assembler and requires the MAC65 program to run.

ACT ACTION file. Requires the ACTION cartridge to use.

TXT Text files created by word processors and text editors. This article is an example of a text file. It can only be read and can't be run as it is not a program.

DOC See TXT. Also called a documentation file as it usually contain instructions on running a program or playing a game. Always read first if on disk for details on loading a disk or program.

ASC See TXT. ASCII text file may be read by any text reader or word processor as they contain no control characters.

ADV Adventure game.

DAT Data files. Required by some programs to store information the program requires. Can't be run or usually read.

FIL See DAT.

FNT Font files. See DAT. Used to store character definitions for printing.

NLO Near letter quality font files for DAISY*DOT printing program

BAK Backup files. Copies of a file made in case the original version in corrupted or destroyed. COPY backup files to the original filename before running or reading. NEVER work on your backup.

TMP Temporary files which contain information that will be needed for only a short time. This are created by some programs when they are needed.

ARC Archived files. These are backups of files which in some cases are compressed to save space. They may also contain more than one file to keep all the files required for a program together. Requires an UNARCHIVER to restore files.

DCM DISK COMMUNICATOR file. See ARC. To use this file you require the program DISK COMMUNICATOR which is a complete disk archiver and compressor.

ALF See ARC.

SCR See ARC. Requires SCRUNCH program to restore.

PAK See ARC.

SUP Supplementary file may be text or other data. Can't be run but may be read if text.

PIC Picture files. Require viewer program.

RGB See PIC.

GR8 Graphics 8 picture.

GR9 Graphics 9 picture.

CVS CVSquash picture. Requires program CSView to restore. May also be CSV.

AMP Music file. Requires AMP music program to run.

MUS Music file. Contains data to play music with appropriate player.

SND See MUS. Sound file.

Next month I'll cover another section of the Library. Remember buying disks helps your club and provides you with excellent software at an affordable price.

Atari ST Librarian Report



By Roger Perrie

Here we are a full year since the last State of the Library Report. What have I accomplished in this year! I'm still re-editing original disks. The old single sided club disks were one hell of a mess. Nearly all of them had to be "re-vitalized". That means to run them through a program that will re-magnetize the tracks on the 'media', without destroying the information contained in these tracks.

Compounding this problem was the use of non-standard disk formats on these earlier Library disks. Some disks are formatted 720k, some are formatted as high as 880k. As the disk media deteriorates, these non-standard disks

failed fast because they were only designed to work with nominal 1 Meg unformatted or 720k formatted disk capacities.

Restoring these disks has been a nightmare in itself. For the most part the early club disks were copies from the various ST magazines that were around at the time. So I have been scouring the fields looking for these old mags. It has proven easier in most cases to find another source for the program than to try and resurrect the disk itself.

Making up the ASCII file telling the user what the file does, is something else to. In a lot of the cases, I actually had to run the program and then write down the author's name and the real name of the program and what the program did. My wife claims that I got even more miserable than I normally am, and grumpier than a bear with a tooth-ache.

I've given up trying to put all of these files into .TOS self-extracting format. I am leaving a lot of them in the .ZIP archive format, so if you don't know how to unzip a file, you will have a lot of trouble with these disks. By saving the file with the name 'PROGRAM.ZIP' and the description file, with the name 'PROGRAM.TXT', it has made it a lot easier to keep track of the files. This gives me two files with almost identical names. The first is the actual archived files contained in their very own folder. The second is it's description file for immediate viewing. No need to UnArc the .ZIP, just read the file marked .ASC.

I have then put these files into theme disks, (programs of similar usage), and put them into the general archives (club disks).

The current master disks will remain in use until all of them have been repaired and the files extracted and archived. Descriptive files made up and stored along with the actual compacted file. When this has been fully accomplished, the files will be put onto disk once again, but in the "THEME" disks! This means that if you buy one of the archive disks it will contain similar programs. It may contain only games or possibly two or more word processors. Due to the way the archiving process works the original ten, single sided 360k floppy disks have been edited down to one single IBM formatted double sided 720k disk. This will be the club standard for future disks. I might add that the disk holds 52 programs of all descriptions along with the ASCII file description and a little blurb about the HBO. The only other thing you will need is a copy of ST-ZIP. Preferably version 2.6, as

that's the version that I used to compact these files with.

If you have trouble with ST-ZIP, it is available on a disk, set up to run from the club library. It has been set up to run on a 520ST with a double-sided disk drive so it should work well for everyone. If your system should differ from this hypothetical setup just open up the config area and change the system to suit yourself.

In the mean time, I am still making up disks for club usage. Unfortunately the print-outs keep disappearing. I will endeavor to get caught up with an up-to-date listing of the latest disks and have that on hand for club meetings. The new CATALOGUE on DISK will be out hopefully before the end of the year.

Midi Notes



By Ron Palangio

The ATARI ST and MIDI

In this continuing series of articles we'll look at the development of the MIDI standard, and cover some fundamental MIDI terms and concepts. I hope to review some different MIDI software applications available for the ST, and show you how to get started with your own ATARI ST MIDI system.

MIDI: A Brief History

MIDI or Musical Instrument Digital Interface is a worldwide standard system which allows electronic musical instruments, computers and other music-related equipment to "communicate" electronically. Before the advent of MIDI, electronic synthesizers existed only as independent units. In 1983, however, two synthesizer companies, Roland and Sequential Circuits combined efforts to develop a system which would for the first time allow instruments to be "patched" or wired together in series.

Under this system musical events produced on one instrument could be transmitted to and reproduced by a second in exact synchronization. Thus, for example, one keyboard could act as a trigger for any number of others patched together with special MIDI cables. Through this additive process of combining sound sources, interesting timbres could be produced easily.

Soon after the development of the system, a paper entitled MIDI 1.0, was published by the various electronic instrument manufacturers. This report explained to the public that through the standardization of the MIDI system, all instruments equipped with MIDI compatibility could be used in combination regardless of the manufacturer. With their rare act of co-operation a truly revolutionary new technical potential was created.

(1) It should also be noted that this cooperation between competing manufacturers continued into the 1990's with the standardization of MIDI patches known as "General MIDI". Nearly every synthesizer sold today is General MIDI. A Roland piano sound for instance may be found on patch number 01, it will also be patch number 01 on a Yamaha, Korg or Ensoniq synth because of General MIDI. This makes it easy to pass a song file between keyboard set ups without too much editing and hassle. MIDI information can be passed between different software, hardware and even computer platforms. You can write a song on a Korg synth MIDIed to an ST, save it as a .MID file and pass it on to PC plugged into a Roland synth.

A plethora of MIDI based synthesizers and hardware developed through the 80's, almost any electronic musical device could now be "MIDIed" up. Synthesizers, drum machines, sequencers, samplers, sound processors, effects, digital delays, small multi-track tape recorders, mixing consoles and even lighting systems all could be hooked up and controlled via MIDI. Non-keyboard instruments could also access MIDI devices, guitarists, woodwind, brass and even vocalists could trigger MIDI information with new input devices and controllers. The computer became the natural central nervous system this new generation of musical equipment.

By the mid 1980's music making had changed forever, computers had become an integral tool in the musician's arsenal. Performing and composing with computers opened new musical possibilities that were only limited by a musician's creativity. In conjunction with small multi-track tape recorders, high quality recording technology and production came into the home. It was now possible to record a professional level product with a computer, keyboard and multi-track recorder in your own bedroom.

The ATARI corporation recognized early on the potential of MIDI and insured that the new ST line of computers would be in on the MIDI revolution. One of the ST's

most identifiable and innovative features were the built in MIDI IN and OUT ports, this made the ST a natural MIDI machine (it still is the only computer that comes with a built in MIDI interface!). By the late 1980's the ST was the computer of choice among a great many musicians, and scores of innovative MIDI applications were available for it. For anyone interested in exploring the world of MIDI the strength of the ST in this area is reason enough to keep your Atari or even buy one.

HOW MIDI WORKS

The MIDI specification begins with a set of common hardware requirements. Each MIDI-capable instrument is equipped with a transmitter and receiver, though certain peripheral devices, such as signal processors, may have only a receiver or a transmitter. The interface operates at a 31.25 Kbaud asynchronous (serial) transmission rate. Each byte consists of a start bit, eight data bits, and a stop bit, for a total duration of 320 ms. per byte. While this was adequate for the applications originally envisioned, it has proven to be one of the major stumbling blocks for generating more complex data streams in time-dependent situations. Though several manufacturers originally included alternative parallel interfaces on their instruments for linking their own peripheral devices, the overwhelming success of MIDI has seen the abandonment of almost all other interconnecting formats.

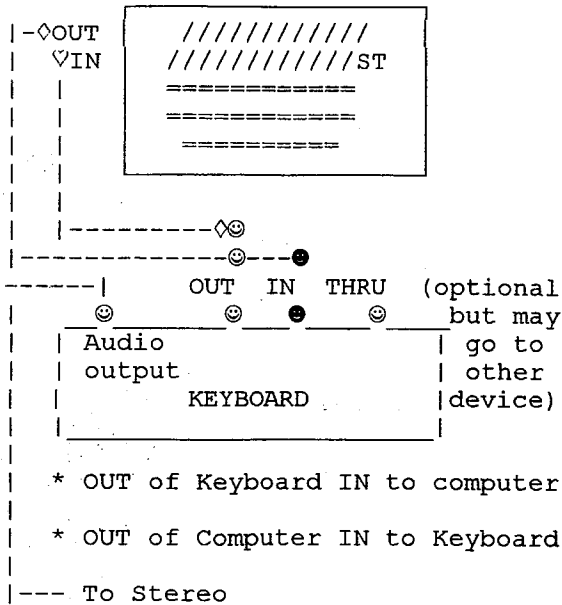
MIDI cables connect instruments by means of MIDI IN, MIDI OUT, and MIDI THRU jacks. The MIDI THRU jack provides a direct copy of data coming into the MIDI IN jack, providing the ability to "daisy chain" several instruments and devices together. Any action on an instrument that corresponds to a particular MIDI code (such as a key depressed, or a program button changed) will normally transmit that action's code to the MIDI OUT, but not to the MIDI THRU.

(2)

MIDI IN - receives information
MIDI THRU - re-transmits information
 - allows you to daisy-chain equipment
MIDI OUT - sends information

A BASIC Computer/MIDI SETUP (for one Multi Timbral Synth)

Computer (ST)



This has been a brief overview of the history of MIDI and the Atari ST's involvement in it's popularity. I hope you've gained an awareness of the potential and flexibility for creating music that MIDI represents. For ST users who fancy themselves as musically inclined MIDI enables you to write, arrange and compose very professional sounding music with ease.

If you have a song in your head and an ST in your midst, your musical ideas can easily be realized.

Next issue we'll discuss MIDI channels, Multi-Timbral keyboards and sound modules, as well as an introduction to sequencing using Notator SL.

* Credits to

(1) Dave Tabone B.Mus. who is teaches a Grade 12 MIDI studies course at Cardinal Newman C.S.S. in Hamilton Ont.



(2) Jeffrey Hass, Center for Electronic and Computer Music, Indiana University

RAM/ROM Control on XL/XE

By John Picken

Reprinted from the January/February 1996 XIO3 The Garden City Atari Computer Enthusiasts Newsletter located in Victoria, B.C.

(Continued from last issue)

Auxiliary ROM Control

Program control of Atari BASIC, OSS Super Cartridges, the R-Time 8 cartridge and, in a limited fashion, SpartaDos X, is fairly simple but for one fact: there's nearly no documentation available on the subject. What I present here is gleaned from a bit of disassembly and a lot of experimentation (pretentious word for "try until it don't crash").

Note that all following references to a cartridge being "present" imply that it is turned on--if it's plugged in but turned off, consider it "absent". You may always consider the R-Time 8 to be absent unless you're actually trying to access it. First let's look at the addresses used for, or in conjunction with, auxiliary ROM/RAM control.

PORTB has already been covered; just keep in mind the function of the BASIC bit.

BASICF is a flag in low memory to tell the OS, on system reset, how to set bit 1 of PORTB. If this flag contains any non-zero value the BASIC ROM will be disabled.

TRIG3 is an address on the GTIA chip which was used for joystick trigger #3 on the 400/800. On the XL/XE it is a cartridge status indicator; if a cartridge is present it reads 1, otherwise it will be 0. There is no other possible reading at this address.

GINTLK is set, on boot, by the OS and is a copy of TRIG3 during the deferred vertical blank interrupt and, if the two don't match, goes into a "soft" lockup (i.e. a reset will re-boot).

CARTCK holds a checksum, calculated on boot, by the OS. On a reset, if a cartridge is present as signaled by TRIG3, the OS re-calculates the sum and compares it

with CARTCK. If the two don't match, the OS assumes you've pulled or inserted a cartridge and immediately re-boots. Note that Mapping the Atari is wrong on this: it applies to all XL/XE's not just the 1200.

CARTCK, TRIG3 and GINTLK are effective for all cartridges except (in part) the R-Time 8. One other important thing to note is that the TRIG3/GINTLK comparison occurs during the deferred vertical blank. This means you can fool around with a cartridge to your heart's content as long as the stage two vblank doesn't occur and you don't hit Reset. You can prohibit vblank2 in any of three ways: disable all NMI's, set CRITIC to a non-zero value or, most simply, use a SEI opcode.

The hardware address range for all cartridge control is \$D500-\$DFFF. Within that page, OSS cartridges use \$D500-\$D50F, the CSS MUX OS uses \$D570-\$D57F, the RT8 uses \$D5B8-\$D5B9, and the SDX uses \$D5E0-\$D5EF. This sounds straightforward, unfortunately it isn't.

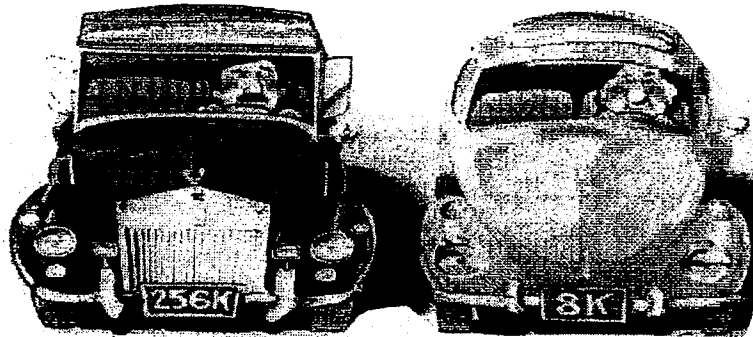
Atari-type Cartridges

I made no mention of Atari cartridges in the address ranges because once you stick one of the beasts into the slot, your only control over it comes with the power switch or by using SDX. An Atari cartridge can not be turned off by software unless SDX is present (even if turned off). SDX can control one because it sits between the computer and the cartridge and can, thereby, zap it electronically. However, the foregoing discussion of TRIG3 and GINTLK remains fully applicable.

OSS Super Cartridges

Though the control range is \$D500 to \$D50F, the cartridge address decode logic is only four bits wide. This means that any access (read from, write to, or otherwise manipulate) a \$D5xy address affects the cartridge which ignores the "x". OSS cartridges react to the whole \$D500 page based on the low 4 bits of the address.

To enable an OSS cartridge bank, add the bank number



to \$D500 and access that address (i.e. for bank n, STA \$D50n, LDX \$D50n, STA \$D500,x where the x register holds n, etc.) In theory, a cartridge should be able to contain up to fifteen 8k banks and still allow you to turn it off. In practice, they contain two or three.

The cartridge bank number is found at location \$AFFF. Valid values at \$AFFF are 0,3 and 4 for Action! and 0 and 1 for MAC/65. Other bank values produce varying results. MAC/65 ignores bits 1 and 2 so

any value from 0 to 7 results in selection of either the odd or even bank. With Action! attempts to select other bank numbers result in selection of one of the real ones or in selection of nothing i.e. a monitor shows a pile of \$AF's in the \$AF page just as when you examine page \$D7 and get \$D7 at all addresses.

Addressing bit 3 turns cartridge ROM off. BASIC XE has banks 0,1 and 9 but bank 9 is RAM. In bank 9, the cartridge is off and RAM under it is being accessed but TRIG3 stays high; a sneaky way to avoid having to worry about GINTLK. The other constant for all cartridges is bank 0 which is the bank in which the cartridge boots and initializes.

OSS Cartridge Examples

Following are several examples of cartridge and BASIC control with SDX not present. I'll start with equates for all examples from Mapping the Atari (XL edition):

```

WARMST -$08
BOOT?  -$09
CRITIC -$42
RAMTOP -$6A
COLDST -$0244
CARTCK -$03E8
BASICF -$03F8
GINTLK -$03FA
TRIG3  -$D013
PORTB  -$D301
NMIEN  -$D40E
EDITRV -$E400

```

Here's the simplest: turn off a cartridge and enable BASIC assuming both are actually present.


```

SEI      ; Kill stage 2 vblank
STA $D508 ; Kill any cartridge
LDA PORTB
AND #$FD ; Drop BASIC bit
STA PORTB
LDA TRIG3 ; This should be 0
STA BASICF ; Flag it and
STA GINTLK ; Correct the cartshadow
CLI      ; Enable stage 2 vblank

```

Now let's access RAM under a cartridge:

```

SEI      ; Kill stage 2 vblank
LDA PORTB
PHA      ; Save Portb
ORA $02 ; Kill Basic ROM
STA PORTB
LDA $08 ; Assume no cartridge.
LDX TRIG3 ; Check assumption.
BEQ GOTBNK ; Go if none or off,
LDA $AFF ; else get the bank

```

```

GOTBNK
PHA      ; Save cartridge bank.
STA $D508 ; Kill any OSS cartridge
LDA TRIG3 ; Set the shadow before
STA GINTLK ; the stage 2 vblank!
CLI      ; Enable stage 2 vblank

```

Do whatever in the RAM, then restore the previous status:

```

SEI      ; Kill stage 2 vblank
PLA      ; Recover cartridge bank
TAX      ; and restore the cart
STA $D500,X ; to it's prior status
LDA TRIG3 ; Reset Gintlk to
STA GINTLK ; correct status
PLA      ; Restore prior Basic
STA PORTB ; rom status
CLI      ; Enable stage 2 vblank

```

Just turning a cartridge off is simple:

```

SEI      ; Kill vblank 2
STA $D508
LDA TRIG3 ; Make sure it went
STA GINTLK ; off and flag it
CLI      ; Enable stage 2 vblank

```

Cold-starting an OSS cartridge is only slightly more complex:

```

SEI      ; Kill vblank 2
STA $D500 ; Enable cart bank 0
LDA TRIG3 ; Set the shadow
STA GINTLK ; correctly

```

```

LDA PORTB
PHA ORA $01
STA PORTB ; Ensure OS is on
CLC      ; Calculate the
LDX #0   ; checksum
TXA      ; for reset.

```

```

CSLOOP
ADC $BFF0,X ; Note the sum
INX         ; includes the
BNE CSLOOP ; first 240 bytes
STA CARTCK ; of the OS ROM.
PLA        ; Restore any RAM
STA PORTB ; OS (or Sparta)
CLI        ; Enable stage 2

```

After cold-starting an OSS cartridge or BASIC, set WARMST to 0 to flag a boot so that the buffer pointers are cleared. If you don't, you can, for example enter BASIC, type LIST and get an endless display of zeros and/or a lockup. Then initialize the ROM. With Action! and BASIC this doesn't matter as the initialization routines just RTS; with BASIC XE I'm not sure; with MAC/65 it's required. To initialize any cartridge

```

LDX $FF ; Say we're on a boot
STX COLDST ; and make sure all
INX      ; flags reflect this.

```

```

STX WARMST
INX
STX BOOT?
JSR INIT ; Go do it
LDX $FF ; Say we're back to
STX WARMST ; normal status, i.e.
INX      ; what happens on Reset.
STX COLDST ; Note that some or all
INX      ; carts play with some
STX BOOT? ; of these flags!
RTS

```

```

INIT
JMP ($BFFE) ; Cartridge init vector

```

After enabling or disabling a cartridge or BASIC, you also have to ensure top of RAM and screen pointers are correct. To do this, execute a "GRAPHICS 0". In machine language terms, you set RAMTOP and then close and re-open channel 0 to the "E:" device. You can do this in the traditional manner via CIOV or more simply by calling the following subroutine with the accumulator holding \$C0 if turning ROM off and \$A0 if turning it on.

```

GRAPHO
STA RAMTOP ; Either $A0 or $C0
LDX #0     ; Indicate channel 0

```

```

LDY #2      ; Point to Close vector
LDY #0      ; and now to Open vector
EDO
LDA EDITRV+1,Y
PHA
LDA EDITRV,Y
PHA
RTS

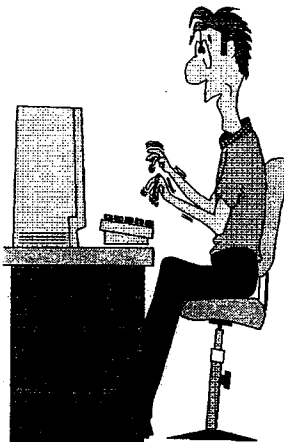
```

Be aware that turning off BASIC XE does not free up the RAM under the cartridge if you intend to later restore the cartridge. BXE uses that RAM as well as that under the OS floating point routines and also (undocumented) sets an I Be aware that turning off BASIC XE does not free up the RAM under the interrupt vector in the last page of RAM (\$FFFx).

One final note on turning ROM off or on: following the Graphics 0, an RTS under Sparta will usually lock up the keyboard requiring a reset. Sparta installs its own E: handler so when you use the OS handler to reopen E: Sparta's vectors are no longer valid. The simple way around this is to exit via a JMP (DOSVEC). This is probably a good idea with any command-processor FMS where you can use a batch file instead of an autorun to set things up.

The SpartaDOS X Cartridge

The SDX boots in bank 0 but works in bank 1 with one subroutine call back to bank 0 via low RAM. You can tell the cartridge had to contain numerous banks just by looking at a directory of CAR: but I have not discovered any single location containing a bank identifier and I doubt there is one as its existence essentially would mean a "hole" in the middle of each 8k ROMdisk bank. There is, as mentioned, a low RAM subroutine to access banks based on an index which I suspect is used to load files from CAR:



CAR.COM uses the following combinations to control all three ROMs in the cartridge area. The values under "SDX" and "OSS" are offsets from \$D5E0 and \$D500 respectively and those under BAS are the value in bit 1 of PORTB. The "on" under the OSS column is the value found at \$AFFF before the cartridge was last turned off and used to reenale it. The \$0C value for SDX is what

causes it to latch any cartridge off and the \$08 makes it transparent so that the cartridge ROM is accessible.

SDX OSS BAS OPERATING CONDITION

SDX	OSS	BAS	OPERATING CONDITION
\$01	\$08	1	in DOS or low RAM
\$0C	\$08	1	in high RAM (X.COM)
\$0C	\$08	0	in BASIC
\$08	on	1	in cartridge

If you're going to play with SDX banks remember that any read or write to a \$D5Ex address will affect an OSS cartridge and TRIG3. Since TRIG3 is affected, GINTLK and CARTCK also come into play. So the example given of how to access RAM under a cartridge needs to be modified if SDX is present. Let's look at a subroutine to access RAM in the cartridge space taking in the possibility of the presence of BASIC, SDX, or an OSS cartridge.

ROMCTRL

```

LDA PORTB
PHA ORA #$02 ;Any BASIC rom off
STA PORTB
LDA RAMTOP ;This might be easy
CMP #$A0+1
BCC NOLUCK ;Not quite
PLA ; See note a. following
STA BASICF ;for an explanation
JMP DOSTUFF

```

NOLUCK

```

LDA TRIG3 ; Is a cart present?
BNE CART ; Yes, go.
JSR DOSTUFF; Still fairly easy
PLA
STA PORTB
RTS

```

CART

```

PLA ; See note a. following
STA BASICF ; for an explanation
SEI ; Kill stage 2 vblank
LDA $AFFF ; Get OSS bank number
PHA ; Save it
STA $D5E8 ; Turn off both carts
JSR DOSTUFF

```

We know a cartridge was on. Now we have to restore it correctly.

```

PLA ; Recover bank number
CMP #$10 ; Is it valid for OSS?
BCS SDX ; No, must be SDX.
TAY
STA $D500,Y; Restore OSS cart bank
BCC CARXIT ; Go always

```

SDX

```
STA $D5E1 ; Enable SDX normal bank
STA $D508 ; Kill OSS cart (note b.)
```

CARXIT

```
CLI
RTS
```

Note a.

The reason for discarding the PORTB entry value is to allow for 512k + RAM expansions. As mentioned previously, the OS doesn't know extra RAM exists and has no way of knowing BASIC does not exist on large upgrades. As a result, it sets PORTB and BASICF based solely on the Option key at boot and uses BASICF to determine which status to restore on a reset. On large RAM upgrades this leads to major problems for programs using extra RAM as the program can end up in the wrong 256k bank. Unless BASIC is actually on, it is always advisable to flag it off and to set its bit high in PORTB.

Note b.

We knew a cartridge was on or we never would have got to that portion of the code. As it wasn't the OSS cartridge, it had to be SDX. But, because the dumb OSS cartridge reacts to the \$D5E1 address, we had to turn it off again after enabling SDX. For the same reason, a single access of \$D5E8 was sufficient to turn both off.

The R-Time 8

Control of the RT8 is built into all versions of Sparta from 3.2 on. As far as I know, all you can do with the RT8 is set or get time and date information. The only problem in doing this is that accessing the RT8 registers will affect an OSS cartridge. Because of this the RT8 has two identical user accessible registers \$D5B8 and \$D5B9. According to the RT8 source, addressing \$D5B8 will turn off a cartridge and \$D5B9 will turn one on. I doubt the latter statement as bit 3 is being accessed. I think, rather like BASIC XE's RAM bank, it turns off the cartridge but sets (or leaves) TRIG3 high.

The RT8 has seven internal registers which work in binary coded decimal. Starting from #0 they are: seconds, minutes, hour, day of month, month, year, and day of the week (#6). Seconds and minutes range from 0-59, hours from 0-23, day from 1-31, month from 1-12, year from 0-99. Day of the week ranges from 0 (Saturday) to 6 (Friday). When you read or write one of these registers

the sequence is always the same:

1. Wait until the RT8 is not busy.
2. Store a value from 0 to 6 into \$D5B8 or \$D5B9 indicating the register you wish to address.
3. Read/write the same address to get/set the most significant digit (the low four bits are the valid data).
4. Read/write the same address to get/set the least significant digit (the low four bits are the valid data).

The source code released by ICD indicates that reading a register should be repeated up to three times accepting two values that match or, failing a match, the first one. When setting a register, it recommends reading it immediately afterward to ensure the value was really accepted and allowing 10 tries.

Here's one way to read and write RT8 registers without worrying about an OSS cartridge. Much of this is from the source released by ICD. In this example, the buffer is set up in the same order as the TR8 registers. With Sparta, you would have to cross refer to the order in which DOS saves time and date and keep a separate byte for day of the week.

*=\$F0 Floating point zero page

```
TEMP1
*= *+1
TEMP2
*= *+1
RETRY
*= *+1
BUFFER
*= *+1 Seconds
*= *+1 Minutes
*= *+1 Hours
*= *+1 Day of month
*= *+1 Month
*= *+1 Year
*= *+1 Day of week

*= WHEREVER
SEI ; Kill vblank2
LDA $AFFF ; Get any cart bank
CMP #$10 ; Is it valid?
BCC SAVBNK ; Yes go, else use
LDA #$08 ; the "off" value. SAVBNK
PHA ; Save cart bank
```

; Verify RT8 present and working

```

JSR READ      ; Get seconds
CMP #60
BCS GLITCH   ; if >59 then error
STA TEMP1
LDA RTCLOC+2
ADC #90

WAIT
CMP RTCLOC+2; Wait about 1.5"
BNE WAIT
JSR READ      ; Read again
CMP TEMP1     ; Same as last?
BEQ GLITCH    ; Yes, not working
SEC
SBC TEMP1     ; Ensure >3
BCS CHECK3   ; It is
ADC #60       ; else did it roll over?

CHECK3
CMP #3
BCC RT8OK    ; Yes, RT8 is ok

GLITCH
LDA # <RT8ERR; Set for error message
LDX # >RT8ERR

EXIT
STA ICBAL
STX ICBAH
LDA #9        ; Print to col
STA ICCOM
STA ICBLH     ; Plenty of length
PLA
TAX           ; Restore cart bank
STA $D500,X
CLI          ; Restore vblank2
LDX #0        ; Select channel 0
JMP CIOV     ; Exit with message

First read the clock regs into the
buffer

RT8OK
LDX #6        ; Point to day of week

RCLOOP
JSR READ
DEX
BPL RCLOOP   ; Change values you want
                ; in the buffer and then
                ; write it back to the clock.
LDX #6

RCLOOP
JSR WRITE
BNE GLITCH   ; Exit if write failed
DEX
BPL RCLOOP   ; else do all 7

LDA # <RT*SET ; Set success message
LDX # >RT8SET
BNE EXIT     ; Branch always

RT8ERR
.BYTE "RT8 Error",155

RT8ERR
.BYTE "RT8 Set",155

; Subroutine: wait til clock is
; not busy or exit on time out.
; Enter: x=clock reg to access (0-6)
; Exit: x unchanged, clock ready,
; and clock register selected

WAITCL
LDY #$FF     ; Timeout value

WAITC
LDA $D5B8
AND #$0F     ; If low nybble=0
BEQ READY    ; clock not busy
DEY BNE WAITC ; Else time out

READY
STX $D5B8    ; Set reg #x to read/wrt
RTS

; Subroutine: read RT8 reg once
; In x=reg#
; Out a=byte x=reg#

READ1
JSR WAITCL
LDA $D5B8    ; Get high byte
LDY $D5B8    ; Get low byte
AND #$0F     ; Convert bcd to hex
STA TEMP1
ASL A
ASL a
ADC TEMP1
ASL A
STA TEMP1    ; Temp1=(height*10)
TYA          ; Add in low byte
AND #$0F
ADC TEMP1    ; Return byte in a
RTS         ; Note c=0 x=x y=trig3

; Subroutine: read a clock register
; and accept best 2 of 3 readings
; or the first if none match.
; in: x=reg#
; out: a=value($)

READ
JSR READ1

```

```

STA TEMP1
JSR READ1
CMP TEMP1
BEQ REXIT
STA TEMP2
JSR READ1
CMP TEMP2
BEQ REXIT
LDA TEMP1

REXIT
RTS
;Subroutine write clock register
;with value stored in buffer offset
;by x. Allow 10 tries.
;in: x=reg#
;out: x=reg#, z flag set if ok

WRITE
LDA #9
STA RETRY

WRT2
LDA BUFFER,X
LDY #$FF ; Convert to bcd
SEC SUB10
INY
SBC #10
BCS SUB10
ADC #10
PHA ; low byte
TYA
PHA ; high byte
JSR WAITCL ; y=trig3
PLA ; high byte
STA $D5B8
PLA
STA $D5B8 ; low byte
JSR READ ; Verify it set
CMP BUFFER,X ; correctly
BEQ WRTXIT ; it did!
DEC RETRY
BPL WRT2 ; Never 0 if failed

WRTXIT
RTS

```

The Multiplexer! Operating System (MUX)

The MUX OS makes frequent access of registers in the \$D57x range. A cursory glance at the ROM reveals it uses the following registers on a read-only basis 1, 6 and 7. Registers 2, 3, B, C and E are accessed as write-only while 0 is read/write. Every one of these addresses will affect an OSS cartridge. I found no indication in the MUX code that it makes any effort to accommodate a cartridge, GINTLK or TRIG3.

While I have managed to work around an SDX cartridge and an RT8 in controlling BASIC, OSS cartridges, and even Atari cartridges (with SDX present), I can see no way of doing so with the MUX OS. I believe the idea with the MUX is that once the plug's in the port, you can't use a cartridge anyway. That's kind of unfortunate as it denies you use of a cartridge and access to built-in monitor. As I don't have a MUX to experiment with, I leave that to someone else.

Cold and Warm Starting and Parallel Devices

If you turn off an OSS cartridge and then re-boot under most versions of SpartaDos, you get a "soft" lockup as DOS will enable a cartridge as part of its boot process in testing for BASIC XE. Simply hit Reset and the computer will boot again with the cartridge on. If you want to reenable the SDX cartridge after having turned it off using the COLD command, the method should be obvious by now. What is not obvious are a few other addresses and quirks in warm and cold starting.

On boot the OS calculates ROM checksums and compares them to ones stored in the OS itself. If these don't match, boot doesn't happen; you end up starting at the Self Test screen and a red bar under the heading "ROM". This can easily occur on a system with a PD because cold starting the computer does not cold start the PD any more than it does a cartridge. If PD ROM is enabled, as for a modem handler on a BBS, and you attempt to cold start, you will inevitably end up in Self Test because the ROM checksum will fail. It is supposed to include the floating point ROM at \$D800, but instead gets a bank of the PD ROM.

Note: PD = "parallel device" Ed.

Finally, if you're just going to warm start, decide whether or not you want to emulate a press of the Reset key. Jumping to the warm start vector at \$E474 is not the same as pressing the key; the vector points past the hardware initialization routines. If you want to ensure you clear out all garbage (left over player missiles, keypress, etc.) you have to use the chip reset vector.

The following routine has varying results dependant on the entry point. To enable SDX, enter at XCART. To enable an OSS cart alone, enter at CART. To just cold start without touching the cartridges, enter at COLD. To simulate a press of Reset, enter at WARM.

```

XCART
SEI ; Always before a $D5xx

```

STA \$D5E0 ; Enable SDX

CART

SEI ; Again, dependant on entry
STA \$D500 ; Enable OSS cart

COLD

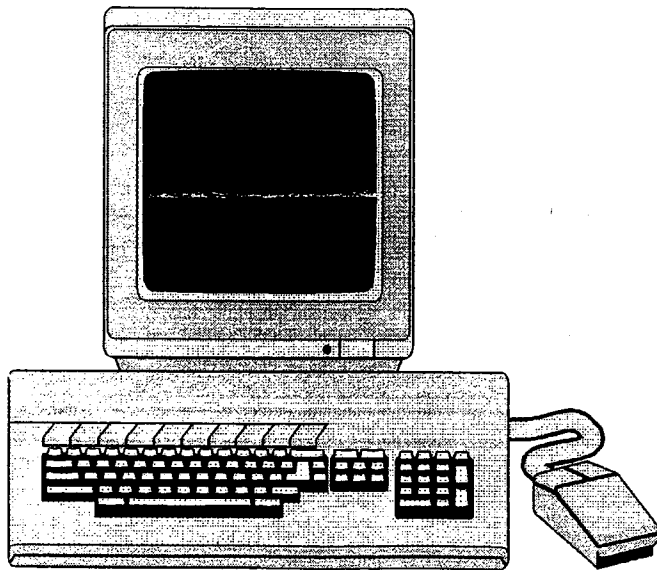
DEC COLDST ; Force a boot

WARM

SEI ; Just in case
LDX #0
STX NMIEN ; Ditto
STX \$D1FF ; Enable floating point
STX \$D1E2 ; Kill MIO RAM (this and
DEX ; following just in case)
STX \$D1BC ; turn off Black box RAM
STX PORTB; Ensure ROM OS is on to go
JMP (\$FFFC); through chip reset
; vector

I won't go any further on this as the Black Box and MIO are, unlike Atari and OSS products, fairly well documented. Whew, when I started this I never thought it would turn into such a monster nor did I think it would take so long to come up with all the ins and outs. Now that you know how easy it is to make use of the RAM under cartridges, under the OS and in extra memory, I look forward to seeing some practical utilities. Here's a few suggestions:

- ▶ Pop-up help screens fro use in MAC/65, Action! or BASIC.
- ▶ An 8k RAM cache.
- ▶ A "pop-up" calculator.
- ▶ A resident DUP.SYS.



H.B.O. Classified Ads

Want to advertise in the H.B.O. Gazette? Submit your ad to the editor! It is as easy as that. This is a free service to H.B.O.A.U.G. members.

For Sale: Atari Jaguar 64 Games complete, NEW in Box \$150. Club Drive complete, \$20 Call 905-544-3387 and ask for Bill

For Sale: Express! Cart terminal. Own one of the BEST terminal programs for the 8-bit in cartridge format. This cart sells for \$50 US funds. Will let go for \$35. Call Bill at 905-544-3387

Incredible Offer! For every article submitted to the H.B.O. Gazette, receive a FREE disk from either library. Submit your article to an executive member today!

For Sale: Atari 800XL, 1050 Disk Drive, Two Joysticks, Paddles, and a box of games for the low low price of \$60 Call 905-544-3387 and ask for Bill

For Sale: Adaptec 4000a controller cards. \$25 each. Call 905-544-3387 and ask for Bill

Wanted: Atari SX212 Modem. Willing to pay \$20 Call collect after 6 pm and ask for Ted Davis. London, Ontario 519-681-8435

Coming In September:

Bob Puff makes his SECOND visit to the H.B.O.A.U.G.

Our 15th YEAR!!!

Does lack of Atari Support Bug you?? Do Something about it! Join the H.B.O. Atari User Group Today!
