

THE ATARI FORUM

A Dialog For Atari Software Developers

Volume I, Number 1

March 1, 1988



A NEW LOOK, A NEW ATTITUDE

Hello and welcome to our new developer's newsletter: THE ATARI FORUM. The new look reflects a fresh and positive attitude within the Atari software group: we want to help you, the software developer, produce higher quality software that is competitive in the marketplace. The newsletter will be published every two months. It is your newsletter; so please do not hesitate to comment on its content, or to submit articles that would be of benefit to other developers.

Along with our new look, we have new goals:

1. We want to improve the dialog between, you the software developer, and Atari—this does not mean that the communication will be one-way: we want to establish a two-way link so that we can mutually benefit from our relationship. Therefore, if there is something you wish to convey that is not right, we are interested in making it right.
2. We want to improve our support program—from a technical, sales, and marketing point of view. Since I will be interacting with sales and marketing, keeping me informed will ensure that your product(s) are known to the Atari organization—please forward any product information—and samples—to my attention. The samples are necessary because it physically confirms the product's existence and allows us to conduct a first-hand evaluation.

I am not completely new to the Atari scene, many of you who have dealt with me know that I am responsive and thorough with my commitments. I have a good understanding of the marketplace as a whole, please don't hesitate to contact me for input and support.

"...We want to establish a two-way link, so that we can mutually benefit from our relationships."

3. Another of our primary objectives is to establish application software guidelines. Specifically, I want to focus on the user interface: that part of the software which the user can perceive. In the next issue of this newsletter, I plan to publish a comprehensive set of guide-

lines; if you have any suggestions, please contact me at (408) 745-2010. Also in the works is a developer's conference. Full details of the conference will appear in our next issue.

I want 1988 to be our best year for the Atari ST; by supporting each other, I see no reason why this can't be achieved.

Joe Ferrari: Editor

Atari's New CD ROM Player

Atari Corporation is pleased to announce its new low-cost (under \$600) CD ROM player for use with the ST line of computers; it is capable of reading both data and audio CDs, and connects directly to the ST's high-speed DMA bus.

The ST is currently one of the best-selling computers in Europe—our goal is to bring that success to the USA. The ST, with its high resolution display, user-friendly interface, and the massive data storage of a CD ROM player, becomes an ideal, low-cost workstation.

But that is only one part of our strategy: The CD ROM player's ability to output audio provides the software developer an enormous potential in the educational field. One of the major weaknesses of existing educational software has been its inability to provide sufficient audio response as part of the machine/user interaction: Atari's new CD ROM player will provide software developers with a new tool for teaching—the possibilities are infinite!

In the entertainment sector, the CD ROM's ability to output audio in stereo will bring a new dimension to video games. It isn't difficult to see some explosive growth in this segment of the market. *(continued on page 4)*

What's Inside

1. ABAQ Page 2
2. Atari on GENie..... Page 2
3. Questions and Answers Page 3

Atari Support on GENie

Last year, Atari began providing online support to our developers. This has proven to be a successful way for Atari staffers and our developers to exchange information.

Since that time, a new service has appeared, providing online support at reduced rates. GENie, a service of General Electric, has developed into a leading provider of personal computer support. In fact, the Atari support areas on GENie are probably larger now than the support areas on CompuServe.

As a result, Atari has opened a new area, for developers only, on GENie. This is in addition to the area on CompuServe.

On GENie you will pay only \$5 per hour for connect charges at 300 or 1200 baud, which is less than half the CompuServe 1200 baud rate. And GENie downloads are usually about twice the speed of those on CompuServe, since the GENie host sends the download file to the local GENie node before it gets to you, effectively eliminating network delays in Xmodem handshaking.

GENie is making a special offer available to Atari developers. You can sign up for GENie service at no cost—GENie is waiving the usual \$18 signup fee. To sign up for GENie, follow this procedure:

Set your computer's terminal program to HALF DUPLEX. Using your modem, dial 1-800-638-8369.

Upon connection, type HHH (and don't hit RETURN). After a couple of seconds, GENie will send you the prompt U#=#.

Type this: XJM11887, Atari and hit RETURN.

Now GENie will walk you through the rest of the procedure, taking your name, address, billing info, etc. When you finish, you will get a user ID and a password, and you'll be able to call your local GENie number immediately.

Once online on GENie, at any menu prompt type M565 to get to the Atari developers roundtable. At that time you'll receive instructions telling you how to get approved to enter this private area.

You can also type Atari at a menu prompt to get to the list of all Atari support areas on GENie. There is a huge ST area there, with more than 4000 files to download and message sections on all kinds of products. Weekly realtime conferences in the ST roundtable, open to the public, are held each Wednesday at 10:00 PM Eastern time. If you are interested in speaking in a formal conference to discuss your product or any subject that interests you, let us know by email to the ATARI address.

If you are used to CompuServe, you'll find GENie to be a bit different. It will take a little getting used to. Feel free to send electronic mail on GENie to the Atari address to get help from us in learning your way around.

We'll see you online!

Neil Harris

ABAQ: A High-Performance Workstation

At Fall Comdex '87, Atari announced a new addition to its growing computer product line: ABAQ. The new computer is based on the revolutionary transputer chip. ABAQ is the first parallel processing, high-performance, personal workstation of its kind.

With ABAQ, advances in applications such as desktop publishing and CAD will be phenomenal. Full page displays will move desktop publishing into another realm. The solid and wire frame modeling graphics capabilities will bring 3-D CAD, rendering and rotation, up to levels that previously desktop computers could not have performed. The better than broadcast quality resolution will allow nearly photographic quality graphics. This feature alone will have a tremendous impact on film, television and video industries ability to computer generate special effects.

The screen resolutions available for ABAQ are very important. All screen resolutions are 60HZ, portrait quality. The highest mode is 1280 x 960 (4 bit/pixel) gray scale. This resolution is excellent for engineering drawings, desktop publishing, film, and television or video special effects work. The second resolution is 1024 x 786 (8 bit/pixel) color. This resolution will be beneficial in any CAD, color picture or graphic work. The 640 x 480 (8 bits/pixel/2 screen) resolution is perfect for animation work. The lowest resolution is 512 x 480 (32 bit/pixel 24 bit/true color plus overlay & tag bits). This resolution may be utilized for finely shaded pictures.

Additional ABAQ advantages include the ability to create a processor farm of upwards of 1000 processors, SCSI drive support, and a floating point processor built into the transputer chip.

The built-in floating point processor in itself is a great advantage. ABAQ offers calculating speeds a single processor workstation can't. The effect of these calculating speeds on any number of applications will be enormous.

ABAQ provides for the addition of up to three internal expansion cards. These cards may include up to 64 megabytes of addressable DRAM, or different graphics cards for specialty applications. The three add-on cards may be configured as 12+1 transputers which would provide 130 MIPS or about 20Mflops in a desktop package. The full bus and appropriate links are available. It also provides connections to parallel processor farms and links to fast peripherals such as a laser printer, disc server, etc.

The ABAQ operating system is HELIOS. A multi-processor, multi-user system, sympathetic to transputer architecture—and familiar to Unix users. The user interfaces include X-Window (Vers. 11) and GEM-VDI driver. ABAQ works with the Atari MEGA Computer.

ABAQ utilizes a high performance transputer microprocessor with a reduced instruction set, capable of delivering computer power in excess of ten times a PC/AT. It is the most powerful single chip computer in the world.

Although the ABAQ transputer chip has a reduced instruction set, it goes beyond the traditional RISC (Reduced Instruction Set Computer). The ABAQ transputer chip consists of a small core instruction set surrounded by collections of application specific instructions. Perhaps its key feature is the ability of the transputer to allow 100 or more transputers to connect together to provide a low-cost desktop computer with the power of a supercomputer.

With the advent of powerful 32-bit microprocessors, and advances in graphics hardware, a new generation of affordable powerful personal workstations has become possible. The systems will, it is predicted, provide an order of magnitude better price/performance than any personal computer currently being sold.

Using the transputer as the heart of ABAQ not only allows the production of a cost-effective advanced personal workstation, it also provides the ability to plug in more power as needed. Such systems set new standards in computing, providing solutions that previously required expensive mainframes, all on one desk.

Questions & Answers

Here are the latest questions from the Atari developers mailbag as answered by John Feagans. Leave questions on CompuServe for PIN 70007,1072 or GO PCS57 for Atari developer SIG information.

1. Corrections

Let us try this one more time... In the last question and answer we tried to show a correct solution to a problem. The Abacus software book, "Atari ST Graphics and Sound", has a typographical error on page 54. There is a C example and a Pascal example of open virtual workstation. The problem with the C example is that this Abacus book does not initialize the `int__in` array as zero-based. The for loop starts at 1. The problem with our fix is that my fingers were out to lunch when I typed in the example. The correct subroutine should be as follows:

```
open_vwork()
{
  int i;
  for (i = 1; <=10; i++)
    int__in[i] = 1;
  int__in[10] = 1;
  int__in[0] = Getrez() + 2;
  v__opnwk(int__in, &handle, int__out);
}
```

2. DOS

Q: I thought that TOS could only detect a floppy disk media change by reading the boot sector and checking the serial number. But if you use a command shell (COMMAND.TOS will do) and type "ls" on disk A: once, then wait for as much time as you want without executing any other commands, then type "ls" again, the directory comes up WITHOUT accessing the disk. Pop the disk and put it back in. The next "ls" will make the OS access the disk. Apparently, there IS some sort of hardware mechanism for detecting a media change. Does anyone know how this can be accessed on a hardware level?

A: The OS twitches the drive select line once every VBI, then reads the write protect line to detect a media change. This procedure does not work with write protected disks. Adding non-Atari drives to the ST makes disk change sensing fail as well.

3. AES

Q: I am looking for information concerning the creation of accessories in assembler. According to ABACUS, to create an application, you must calculate the program

size, and do a setblock call to return leftover memory. Is this necessary when you create an accessory? If not, what should it be replaced with?

A: When the desktop goes to the accessories initialization code, memory has already been shrunk to fit. You still have to malloc buffers and allocate your stack. That is why we provide two start modules in the Atari developers kit.

Q: My company is considering developing a 16 bit sampler (audio) for the ST along with support software. I was wondering if you might be able to help me out with some questions I have concerning the connection of any hardware we might develop to the ST.

1) Is there any particular advantage to using the cartridge slot over the DMA port?

2) Will either or both of these ports be available on the Mega ST and future ST's?

3) Is the ROM expansion slot strictly Read Only? I seem to recall that some cartridge products that I've heard of, both read and write to the port.

4) Are there any problems with daisy chaining devices to the DMA port? The hard drive doesn't seem to have any facility for daisy chaining. Actually, I'm relatively inexperienced at hardware level programming so I'd appreciate any help or sample code that shows how to poll for bytes at the DMA port or code an interrupt handler for it.

5) We're going to need a fast clock for sampling (at least 44 Khz). I was planning on looking at Timer A of the 68901 for this. Will this work or am I barking up the wrong tree? Do you have or know of any sample code showing how to obtain a timer of this nature?

I must apologize for the length of this note and the barrage of questions herein but I'd appreciate any help you can give me to get me started looking in the right direction(s). Thanks in advance.

A: The ROM cartridge port and the DMA are present in all current and in future mega-STs. I recommend using the DMA port because people making cartridges do not provide for any stacking. The cartridge port is read only, but there are ways to write data by using the address lines. Current hard drives do not have any daisy chaining capability but new production drives have two connectors. All hard drives have an addressable unit number. Timer-A is reserved for use by applications.

*Statement: Yes! I agree, John, enough memory must be leftover for darling GEM niceties. TOS is not very friendly when it doesn't have enough memory. I had a program give me an address error (vector 3) from WITHIN the roms. I eventually (after two days) tracked it down to a vs_clip I was using after a fsel__input. Huh? That really threw me because that's a *real* basic operation. However, I had inadvertently let another zero creep into my Malloc() and was stealing lots more memory than I needed. Rather than pass me errors, GEM decided to bomb. Oh well, live and learn.*

A: You should never, on any system, be it ST or another, Malloc memory without first doing a Malloc(-1) to see how much is available. You also must leave memory for resource files that you may request the system to load and also the file selector. The precise amount of memory that the file selector input requires is \$a00 (2560) bytes. You should also leave memory for any program that yours may Pexec().

Q: I have trouble with event library calls after I have done anything with the VDI calls to determine mouse position and button state. It seems like the event library never sees the button come up and thinks it is still down. You have to click again to get the correct state.

A: We do not recommend using `vq__mouse` if you are planning on doing any event library calls such as `event__multi`. What is happening is just as you have described—the `vq__mouse` is siphoning off some button states which are never passed on to GEM. The AES only knows about the last state it had such as that the button was down. There are two acceptable ways around this. First, you can do the exchange mouse or button vector calls in the VDI and insert your own handler which will scan the mouse position and button states and deposit this information in your own variables before passing on to the previous handler. Second, you can query the line-A variables for button state and mouse position which can be accessed by the following example:

```
long linea__vars;
int *gcurx,*gcury;
char *gbutton;
/* initialization code */
linea__init();
/* code to use mouse data */
gcurx=linea__vars-602;
gcury=linea__vars-600;
gbutton=linea__vars-348;
```

```
;68000 assembler to do line-A init
__linea__init:
    movem.l    d0-d2/a0-a2,-(sp)
    dc.w      $A000
    move.l     a0,__linea__vars
    movem.l    (sp)+,d0-d2/a0-a2
    rts
```

Q: I've got a couple of editable fields on a dialog box, I've got an I-BOX defined as exit default, to be able to handle the return key my self, that is: when the user edits the first field and presses return the application is supposed to:

- 1 read the 1st field input.
- 2 fill the 2nd field with some feedback.
- 3 let the user modify it.

I can handle most of it, the main problem is to know in which field is the user when he presses the return key, because I've got to do some other stuff when the 'Ret' is pressed on the 2nd field. Any ideas??

A: You need to go through the object list and see which object state is selected. That's is how you tell.

Q: I wanted to detect a double click NOT in a `form__do` but after receiving a mouse button event from an `event__Multi` call. I can detect a single click and wait by using a timer/button event. This way if a button down event occurs it would be the second click in the double

click. otherwise it would be a single click or a mouse drag. The problem is that the delay causes an undesirable interruption in the mouse operations when there is only one click. The wait for a second click causes a delay when only one click is pressed. Any solutions!!! Please!!!

A: This answer appeared in ATARIDEV on Compuserve from Corey Cole. See his program `Button.c` in DL0 for more information. "I handle double-clicks by requesting single-click through the `evnt__multi`, and keeping track of state transitions myself. This also allows reasonably clean handling of the right mouse button. This approach is a little strange in feel to users accustomed to having to double-click very quickly, however – the second click might be missed if too fast. (On the other hand, I don't have a time limit for multiple clicks in one place, something I prefer to the GEM approach.)"

Q: Does the Alcyon C support floating point?

A: Yes. The one shipped in the developers kit supports single precision with the Motorola fast floating point routines and IEEE single and double precision is available in the optional 4.14 upgrade.

4. New On Compuserve

In data library 7 (for registered Atari Developers only) in the Atari Developers SIG on Compuserve, the following files are new this month:

madmac.arc

A fast macro assembler, also generates `.prg` files directly from modules without requiring linking and or `relmod`.

aln.arc

A fast linker, use as a replacement for `link68`.

Atari's CD ROM *(continued from page 1)*

A CD extended BIOS is being developed to provide ST applications with a standard interface to the CD ROM player command set. In the future, Compact Disk Operating Systems will become available for emerging CD ROM standards (such as High Sierra Group format).

Join us in supporting this revolutionary product. The Atari CD ROM player will be available for software development in February of 1988. Product release is scheduled for the first quarter of 1988. We are interested in working with your company to provide a variety of CD ROM applications—and also to insure compatibility of our products with yours.

We also look forward to exploring possibilities in the MS-DOS world with our line of PC-Compatibles. For more developer information about the Atari CD ROM player, write to us at our Sunnyvale address, Attn: CD ROM Dept. Please send complete information on your company and its products as well. This will be a great help in our efforts to understand and support the CD ROM community.

Mike Schmal

Director of CD ROM Technology