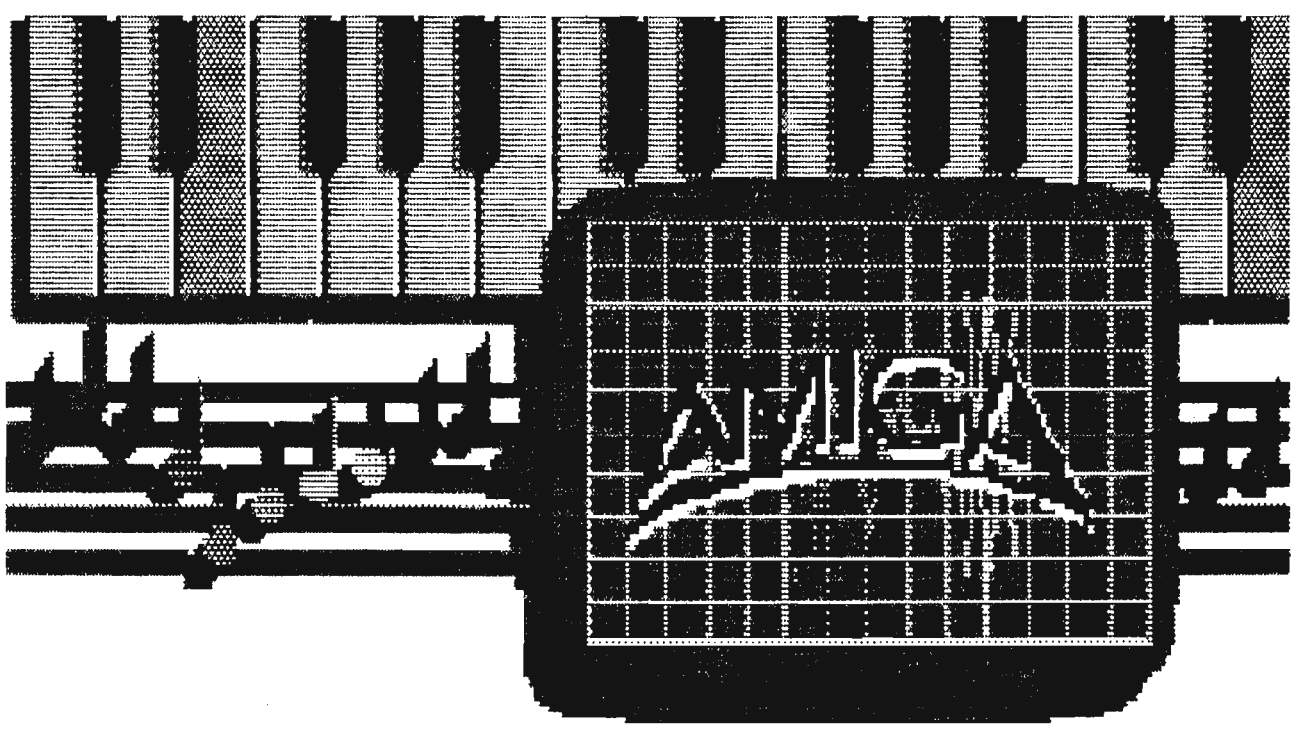


Vol.2 #1 Sept/88 

**SOUNDS LIKE ANOTHER GREAT YEAR AHEAD!!!**



**1st Anniversary**  
1988

# CONTENTS

Volume 2 No. 1  
September 1988.

## FEATURES

My Year of Living Amigaly.....	5
Library Update.....	4
68000 Mnemonics.....	11
AmiExpo Report.....	7

## COLUMNS

The Amiga.....	9
The Amazing Amiga.....	6
Hints & Tips.....	10
Editorial.....	1

## REVIEWS

SuperStar Ice Hockey.....	2
Sub-Battle Simulator.....	3
Rocket Ranger.....	3

## ADVERTISERS!

The BAUG Newsletter is currently accepting advertising of any size for very reasonable rates. Please contact Syd Bolton for further info!

# B.A.U.G.

Newsletter

Copyright (c)1988 Brantford  
Amiga Users Group, All Rights  
Deserved.

Publisher: Syd L. Bolton  
Editor-In-Chief: Syd L. Bolton  
Art Director: Gene Chmielowiec

### Contributing Authors

Robert Beechy  
John Bos  
John Callsen  
Gene Chmielowiec  
James Hassett  
Pat Levell  
Ken Sahagian  
Steve Spedding  
Richard Wildey

### Printing and Reproduction

Hurry Print

### Official Coffee for Syd

Nabob

### Computer Hardware

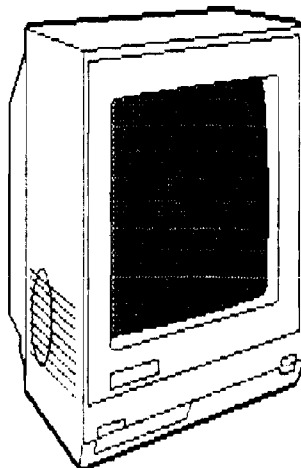
Amiga 1000  
Megaboard 2 Meg Ram Expansion  
Epson LQ-800 Printer  
Digi-View

### Software Packages Used

PageSetter 1.1e (Gold Disk)  
Deluxe Paint II (Electronic Arts)  
SuperBase (Progressive P&S)  
ClipArt #1 (Magnetic Images)

The BAUG Newsletter is an independent journal not connected with Commodore Business Machines. The BAUG newsletter is published monthly (sort-of) by the Brantford Amiga Users Group, 25 Frontenac Ave, Brantford, Ontario, Canada N3R 3B7. (519)-753-6120. Entire contents Copyright 1988 by the Brantford Amiga Users Group. No part of this publication may be printed or otherwise reproduced without written permission from the publisher. BAUG makes every effort to assure the accuracy of articles, stories, and reviews published in the newsletter. BAUG assumes no responsibility for damages due to errors or omissions.

# From the Editor's Terminal



By Syd L. Bolton

Oh well, the vacation is over. **WELCOME BACK!** A big welcome to all of you BAUGsters (thanks, Jim, for the name) back for the second year. I can hardly believe that we are an entire year old! Well, let's not just gleam on the past, but look forward to the future!

Hopefully, next meeting (October) we will be having guest speaker Steve Spencer of Commodore Business Machines down to give a little talk on things. Steve graciously visited us last November, and brought his Amiga 2000. He was well received and everyone had a good time. At this time we are unsure as to what Steve will be talking about, but he's a good public speaker, so you should have a good time regardless. It's nice to see Commodore Canada so supportative of their user groups.

Speaking of support from Commodore, Steve sent me a two-page letter concerning a national "Friendship" day where user group members take their machines to the local Hospital and let the sick-kids get a taste of the Amiga. It runs October 1 and 2. If you are interested in partaking in this event, please get in touch with Gene Chmielowiec, coordinator of this event. If enough response is given we will approach the Hospital concerning this. Act quick as this is a "cutting it close" event. October 1 and 2.

Perhaps the next newsletter will be quite a bit different. I may be using a different computer believe it or not, because of a particular piece of desktop publishing software. "Publish It" is an excellent program that provides extremely high quality output on a 24-pin printer (or even a 9-pin printer!). Look for a new format change soon.

We've got quite a few good things in this issue. Read up about the AmiExpo report. I went to Chicago with my best friend Adam White, and we both had a great time. The hotel was lovely, the deep-dish Chicago-style pizza was *deep* and I drank a lot of *Coca Cola* (there's an inside joke there, relating somehow to alcohol). Anyway, check that out. Jim Hassett has an interesting overview of the past year with his Amiga, and Gene Chmielowiec looks back on the summer and the months ahead. John Bos supplies us

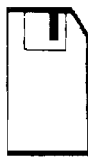
with more technical information, with a 68000 chart and an article on the stack. All in all, a good issue. Enjoy.

Before I go, I must mention a few changes! The fee structure for the club has changed. Yearly memberships are still \$20, however, the newsletter now costs \$1.00 an issue and disk copying is now \$2.00. However, keep in mind that writing an article for the newsletter gets you a free copy of the issue your article appears in, and a free copy of any PD disk in our library. Hopefully, this will promote more user involvement.

And finally, in closing. Some of you may be aware, but Adam White and I have resigned our positions as TPUG AMIGA LIBRARIANS. Are reasons are both personal and "corporate". Frankly, we didn't like the way the club was being run, although we weren't about to step in and help in that area: we had enough problems of our own. Also, it's difficult for me, living in Brantford, to do work for a Toronto-based club. We did our best, and served for a year and a half. However, everything must come to an end. Just like this editorial.

Handwritten signature of Syd L. Bolton.

# Reviews



+



+



=



## **SUPERSTAR ICE HOCKEY**

Reviewed by James A. Hassett

To get into *Superstar Ice Hockey* you must enter a word from the manual as requested from the computer. In doing this you have exceeded the level of effort worth expending to play this game. The shame of it is that the game has the potential for being very exciting but for a misguided sense of realism.

As in real hockey there are two levels to the game. There is the action and the strategy. To play the strategy you must set your lines and trade players although the latter requires that several games be played so as to acquire trading points. During the game you have the option of changing lines and play tactics or letting the computer do it.

The above features are all well and good but the attraction of hockey is the actual moving of the puck. It is at this level that this program fails miserably. Firstly, the graphics are very good and are at their best when a player is checked or takes a shot. You can only control the centre or the goalie, however, and are committed to your choice for the duration of the game. This means your level of involvement is pretty low. Your centre moves quite slowly and can barely be identified by a near invisible blue line under his feet. Things improve slightly when you reduce the size of each team to two players but this ofcourse is not as realistic.

The real killer feature is the fact that if your centre gets penalized you are out of

the game for two minutes. This is alright in real hockey since it gives you a chance to catch your breath but it is unthinkable to remove the only player from the game leaving the computer to play with itself.

The only saving grace of the whole program is the two player option. When used with two man teams it could be interesting. You must ask yourself how often you plug in two joysticks for any game.

My heart felt advice concerning *Superstar Ice Hockey* is to save your money for a good spreadsheet program. You're sure to have more fun.

**SuperStar Ice Hockey**

\$59.95

Mindscape

## **SUB BATTLE SIMULATOR**

Reviewed by James A. Hassett

If you've ever watched Clark Gable or John Wayne wreak havoc on the Japanese Imperial Navy from the control room of a submarine and found your palms damp and your heart all a flutter, or seen, from a standing position, Jurgen Procnow, by sheer strength of will, desperately drive his German U-boat through an allied convoy, because you were too excited to sit; then have I got a program for you.

The *Sub Battle Simulator* by Epyx is a sophisticated, well thought out game (and I use the term game with the utmost respect) that keeps you involved while bringing out the flavour and action of those old movies (which,

after all, is all any of us have to refer too).

SBS boots up to its only screen which, except for the odd specialized window, gives you all you need to play for hours. There is no need to remove important graphics to perform other tasks. The top left area shows a map to aid in navigation and hunting for targets. It indicates your sub, any enemy vessels and sometimes friendly targets such as downed fliers. The map can be zoomed in and out to seven, thirty-five, one hundred and seventy-five and one thousand mile settings. When you get to within seven miles of your target the map can be replaced with a side view of your sub which shows its depth as well as neighbouring ships.

To the immediate right of the map are the sub control tools such as dive/surface, diesel/battery power, and your choice of sonar, radar, periscope, binoculars or tower views. All the views are displayed in the top right corner.

The bottom of the screen contains, from left to right, the depth gauge, speedometer, weapons controls, heading gauge, and heading gauge for whatever view you have selected. All the gauges double as controls by moving the little white "bug" to the desired setting. Your weapons are forward and aft torpedoes, a deck gun (cannon) and an anti-aircraft gun which can also be used against shipping.

The final permanent area of the screen is the centre scroll which displays in text messages from your crew. These include damage you've inflicted on the enemy or that he has inflicted on you.

**B.A.U.G**

These messages are written in military-like language and always refer to you as "Sir".

Radio messages such as position reports and new orders can be sent and received easily. You hear the message as morris code and see it written in a window that you click off when the message is complete.

Actual Submarine operations took many days but the realism of this simulator is maintained by a Time compression device. You can choose one second of real time to be equal to five seconds, thirty seconds, ten minutes, or four hours. If you are on patrol it might be hours or even days before you sight the enemy so you run on the four hour level. When look-outs sight a convoy the time compression is automatically reduced to thirty seconds. All this means that you get a real feel of what it was like to command a submarine while still playing quickly in real time.

This is not just a shoot'em up game. You must become aware of your boats strengths and weaknesses and learn the principles of diesel-electric power if you are to complete a patrol successfully, and your boat is far from invulnerable. You can select target practice which immediatly puts you in range of an enemy convoy or start on a patrol during which you may encounter convoys or be called upon to rescue downed aircrew among other things. When you do open a patrol mission you can choose to be a German or American captain and select what year of the Second World War the patrol will take place in. At the conclusion of your mission you will either be given a summary of your performance and be rated accordingly or your family will get a telegram of condolence from the navy.

**Sub Battle Simulator** is very engrossing and invites play for hours. It excersises your brain while remaining exciting and giving instant gratification

when your torpedoes find their mark.

<BAUG>

**Sub Battle Simulator**

\$54.95

Epyx

## ROCKET RANGER

Reviewed by Syd L. Bolton

When I first bought my Amiga in November, 1986, I was entranced by the most incredible graphics I had ever seen in my entire life. I picked up the box next to the computer and it was labelled "*Defender of the Crown*". Whoah. This was the first piece of software I had purchased.

At the time, *Defender of the Crown* was state of the art - nothing had matched it nor come even close. Over the last couple of years, **Cinemaware** has released several other titles, each better than the last. *King of Chicago*, and *The Three Stooges* were all small steps forward in an increasingly complex game market. Then came *Rocket Ranger*.

This game is more than a small step forward, it's a giant leap. From the great attention paid to detail in the title screens, to the excrutiatingly fine points of game design, this one has it all. The graphics are unmatched in any other title; both in style and the animation associated with such. No skimping has been made. When a character moves, it truly looks like it is moving. The sound is also notable. There are several themes included, each relating to what is happening in the game. I am constantly

hearing different music as I progress, and it is also nicely done in stereo, with great sound effects and incredible speech. Now on to the game.

"A hundred years ago, the Nazis won World War II. You are going to change that". This is basically the setting of the game. You, as Rocket Ranger, are on a lone mission to save the world from socialism.

The world at this time, is surviving on a substance called "lunarium", which can be found on the moon. Your rocket pack is fueled by it, and so are rockets and machines to help in the aid of man's destruction. Given a limited supply you must figure out where to get more on Earth, and eventually, find 5 pieces of a rocket, so that you may assemble it and go to the moon.

Sound easy? Well, of course it isn't. You have five agents who are searching for lunarium, and causing resistance to the Nazi movement. Unfortunately, these agents can often be battered by the enemy.

This game, like other Cinemaware titles, evolves like a movie. It goes from chapter to chapter. If you screw up in one chapter, some fictional explanation will be given (for example, the girl you were supposed to rescue from being fried escaped and is somewhere else now) and you will be allowed to continue on in the game.

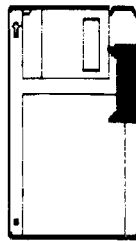
This game uses a fairly unique form of copy-protection. The disk themselves are not copy-protected (so you can make backups) but to go from country to country, you have to tell the computer the proper amount of lunarium, and this can only be obtained from a wheel. Be careful, as the wheel cannot be photocopied and if you lose it, you

are stuck with a game you can't effectively play.

Overall, *Rocket Ranger* is the best Amiga game I have played to date. It was 5 man years in development, is 2 disks long, and requires an external drive. 10 out of 10. <BAUG>

**Rocket Ranger**  
\$59.95

Cinemaware Corporation



# LIBRARY

# UPDATE 0.

By Syd L. Bolton



Over the summer our library of freely distributable disks has increased quite a bit. With the new year, a new fee for copying disks has been installed. It is now \$2 to copy a disk. This is still reasonable and I hope it does not deter anyone from buying the service of copying disks from the club. This, after all, brings you the programs you need and the money that the club needs. It's a circle, just like life.

Any questions or concerns about the library should now be directed towards Gene Chmielowiec, who is our appointed librarian. Now on to the good stuff.

## FISH DISKS

The Fred Fish disk library has now expanded up to disk 146. We have just about all 146 disks in the library, with some gaps which I hope to fill as soon as possible. Please let Gene know if there is a disk you need that we don't have. Thanks.

## AMICUS DISKS

The Amicus collection of disks has now grown to 26. Again, there are a few gaps in the library that at present I am unable to get filled, but I will keep on trying with my distributor.

## TPUG DISKS

At present, I am working on two more disks for TPUG (my last two) but the present size of the library is 14 disks. (A)TAA-(A)TAM, plus a special show disk.

## MISCELLANEOUS

I found a stack of about 80 disks that are not really from any one source, but are just various collection of music, pictures, and programs. Some of them are very good. I did notice things like "BEST OF AMICUS" and "GOLD FISH" which we already have. However, I will give you a sampling of these disks and we'll figure out a numbering system for them.

ANIMATOR: APPRENTICE DEMOS. Including "HURL" and "TREX".

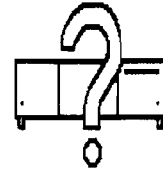
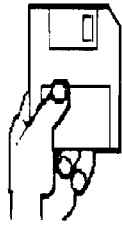
BEST OF AMICUS #3. WBLander, LPem, MacGag, Hockey, Munch2, Target, World, Piano, Journal, POPCLI III, StarChart, ASLClack, GOMF, EXTRA HALF\_BAITE.

MONDO AMIGA ANIMATION. A very interesting demo that includes quarter screen animations running concurrently including "El Gato" (running tiger), the "Juggler", and "Balet." Very interesting. Requires 1 megabyte to run fully.

SONIX SCORES. Heavy on the Woodway, Break It, WildWest, Pass 1, Take it Easy, Sleeping Children, Rainbow, Trance, Poor Ghost, Unknown Space, Nobody Knows Him, Voice After Voice, Up in the Sky, Hi Mabuse, The Law, The Sphinx, Danger Planet, Here I am, Welcome, On The Mars, Sunrise, The Last Fighter, Blood, Black Dog, Dark Night, Sign of the Dead, It's Not Too Late, He is Crying, Electric Dream, The End, Tonight, Strings, Liveless, No One To Help.

<BAUG>

# My Year of Living Amigaly



By James A. Hassett

It hardly seems like a year since I took the rather unnerving plunge into the mysterious world of Amiga ownership. I had been using a Yamaha CX5M but, disappointed with its lack of power and awkward implementation, made the hard decision to spend what for me was a great deal of money.

When I got my new treasure home I poured over the various manuals and tried the tutorials and simple procedures I realized, after some minor frustration, that this was indeed the user friendly tool I was looking for. Now, after using my Amy for these past dozen months it is time to look back and see if this pile of processed sand, crude oil and metal has lived up to its promises and my expectations.

I have never been interested in writing programs or doing anything with the computer for its own sake. Instead, I view it as nothing more than a very useful tool for the pursuit of my modest artistic interest. Because of this, simplicity and ease of use was of primary importance. In this regard the Amiga gets high marks. If I ever meet the person who came up with the idea of icons for function control I'll kiss him full on the lips (unless he's ugly then I'll give him five dollars). The icon system is perfectly suited for use by a computer illiterate like yours truly.

On the other hand, should an occasion arise when I am required by a program to use the keyboard, it is designed with good ergonomics and a logical layout so I am able to type at the horrendous rate of 110 words per hour.

For some one on a budget the Amiga's expandability is just what the theoretical physicist ordered. I can't wait for the recently announced module that will allow the Amiga, when plugged into the cigarette lighter, to drive your car. Until then I shall happily use my extra drive, my MIDI converter, and my memory expansion.

The standard IFF file format is such an obvious inclusion to the design that I can hardly imagine any computer system not having a similar set up. I certainly would not have the collection of software that I do without it.

Now, lest you think this article is low grade advertising copy, allow me to tackle some of what I think are the drawbacks with the Amiga system.

As I have stated in previous ravings ... er ... articles, the manuals that comes with the machine and most of its software are poor. Some can get by with the thinnest of pamphlets. I however need all the help I can get.

The mouse controller is invaluable as most Baugsters would agree, however, from the beginning I have had trouble finding a comfortable position to hold it. It seems to me that a rounder shape, perhaps with the buttons on the side would have been more ergonomically acceptable.

Most of my complaints are directed towards the Amiga software. Programs are part of the system you buy after all. Now each program must

be judged on its own merits but there are some common points to which I address my comments. My patience is wearing thin with the self-indulgent practise of including complicated title screens and theme songs in the program boot up sequence. It amounts to another layer of fancy packaging and I can't help wondering what that effort and wasted memory could be used for in the program itself.

One feature I would like to see implemented is a standard ITF system. That is, and Inter-program Tool Format. Like the IFF this would require all like programs to have standard tools and similar icons for those tools. This would be very important for example, in art/graphics programs. To some extent it exists already but not to the level I would like to see. (See above) I know the programmers among you will scream about limiting their creativity but, lets face it, the home computer industry exists to satisfy the needs of the average user not programmers. If your really good you'll work around such limitations.

That is my first year report on owning an Amiga. I feel that I've been able to progress further musically in the year than I have for many year previous to now. I have created pictures have garnered some favorable comments, and, for the school boy who was judged by his teachers to be totally inept both artistically and musically, this is worth the all the cost in cash and time.

All that I can say is good luck with your Amiga and may you find the satisfaction using it that I have.

<BAUG>

# The Amazing Amiga

*A Big Welcome Into the New Year*

By Gene Chmielowiec

To begin with I'd like to say welcome back to all the members of BAUG on this the first meeting of our second year. If we have any new people joining us this evening I extend a heartfelt greetings to BAUG and congratulations on your obvious good judgement and taste in deciding that the Commodore Amiga is the computer for you.

I hope everyone's summer was a pleasantly relaxing and enjoyable one, and that you all managed to sit down and get some more hands on experience with your computers. But if you didn't, well thats OK too. After all it was summer, and frankly I really didn't expect that too many of you would spend your time indoors on your machines, what with all the beautiful weather we had since our last meeting. To be quite honest, I too managed to spend only a little time on my Amiga these past few months. But at least it was time spent productively.

I've managed however over the summer to upgrade my system so now I have a second 3 1/2" drive, as well as increasing the number of Fish Disks which I have in my collection. To this I should also have a 5 1/4" drive hooked up to my system by the time you read this article. And if all goes well, it looks like Santa (or Mrs. Claus) might even be kind enough to leave a printer under the Christmas tree this year for ol'dad.

But before I get too far ahead of myself I'd just like to repeat what Steve Sped-

ding said in last December's newsletter when he talked about the joys of having a two drive system, in his "Two Drives Or Not Two Drives" article. I can't emphasize more what a pleasure using your Amiga will become once you have aquired a second drive. Mine doesn't even feel like it's the same system. It makes that much of a difference when using your Amiga! Especially if like me, you find yourself constantly loading up either one of the Fish disks, or one of my art data disks. No longer do I find myself faced with the drudgery of having to swap repeatedly between either my Workbench and Fish disks, or my Deluxe Paint and art data disks. It now is one fluid pleasant process. I can't believe I managed to do all the things I did this past year on only one drive. I know that a second drive is an added expense which most of us can't afford at the time we purchase our systems, (myself included) but one (now having gotten one) I feel is a justified necessity, which you should seriously consider at the time of you purchase your Amiga.

Now let me address myself to all of the budding journalists in our group. I know that the number of people in our club who will most likely be contributing articles this year will be high (hint hint) I none the less welcome the opportunity to see and hear anything, and everything that you have on your mind.

Speaking of hearing from you, I have come up with a

novel little way of proof-reading any of the articles which I submit for our newsletter.

What I do once I have typed in and saved my article is to then reboot my system and then insert my Workbench 1.3 disk into my df:0 drive and my saved "BAUG Article Disk" into my df:1 drive. Once my Workbench 1.3 icon pops up I click on it and then go into the "Shell" program that's found on this Workbench disk. What I do once I'm in the shell window is to type the following-

**mount speaker:**  
**copy df1: Amazing speaker:**

What happens now is that the the Shell program after having mounted a speaker device into it's routine, has been instructed to then load my article which in this example I have saved under the name "Amazing" and will then cause my Amiga to recite my article back to me, rather than my having to proofread it myself. (*Editor's Note: I still had to edit it, come on Gene I want the computer to do it all!*) All I do now is to just sit back and listen to my article. This can be a quick and easy way to hear mistakes in your article, especially if like some of mine, they take on the proportions of a novel (like War & Peace).

I hope this method can help you. Come on and submit something. Last year was a fantastic year for our group, and I know this year will be even better with your support and input. <BAUG>

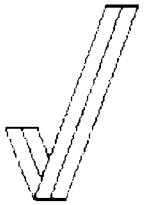


# AmiExpo Report

A Report From the Chicago Show, July 22-24, 1988



By Syd L. Bolton



Large crowds, the big city. Tall buildings, many cars. Deep-dish pizza. Several ways of describing Chicago. All true, too. This summer I had the fortunate experience to attend *AmiExpo* in Chicago, held at the Hyatt-Regency Hotel. Quite an experience. Here's a quick look at what went on there.

At this, the third *AmiExpo*, and the largest, many companies attended demonstrating their new products. When you walk in the door, the NewTek booth is there, with lots of neat video products on display. There are hordes of people, buying up product, examining product, ordering product, trying product. The show went from Friday, July 22nd to Sunday, July 24th. Lots of important people were there, many speaking in conference rooms. I met people like Jay Miner (designer and "Father" of the Amiga), Leo Schwab (famous for his screen hacks in the public domain, and always wearing a cape), Dale Luck (former employee of Commodore-Amiga, a system-software designer), and Mark Brown (editor of *INFO* magazine).

Saturday, July 23rd, was a special day. Ironically, three years ago to that date, the Amiga was first unveiled in New York. To celebrate this, *AmiExpo* and *AmigaWorld* held a birthday party, complete with cake (that tasted to me more like sawdust). Jay Miner cut the ceremonial first piece. This huge cake

was shaped like an Amiga. Of course, there were many other cakes, to feed the jam-packed room.

I have special memories of this birthday party. Birthday cards were scattered around the room, to be signed, and sent to Commodore. Two mementos I have from this party are "Amiga" balloons. Just don't tell anyone I stole them.

On to the good stuff. There would be way too many product announcements and so on to cover in this article (or even a full issue of *Computer Shopper!*) but I'll give you a highlight of some of the companies and products that they were demonstrating.

**NewTek** was there in full-force, handing out NewTek screwdrivers, pocket protectors and the like. Version 3.0 of their popular digitizer software, *Digi-View*, was shown. The new version includes overscan support, allowing you to digitize right into the borders, effectively increasing your digitizing space by 30%! It also includes half-brite support to allow screens with 64 colors, if your machine has it (only Amiga 500s, 2000s and late-model 1000s have extra half-brite. If you have an older 1000, contact your dealer about getting the upgrade Agnus chip).

NewTek also demonstrated a working version (although still in wire-wrap stage) of their "Video Toaster". This is

the ultimate in video hardware. It provides digital video effects, including real-time video manipulation, broadcast quality special effects, smooth picture movement, pre-programmed real-time video effects like page flips, turns, spheres, blinds, moving blinds, pixelizations, fish-eye, mosaic, montage, pushes, splits, transpositions and more! It has a frame grabber built-in, that grabs images in 1/60 of a second, and a frame buffer which allows compatible Amiga software to run in millions of colors! Finally, it is a broadcast quality genlock, providing RS-170A output. Add-ons which are coming soon include real-image processing support, TMS 34010 video coprocessor, professional quality chroma-key, professional NTSC paint program (allowing millions of colors) and a programmable video switcher. Whoah. And it works on all models of the Amiga, at a retail cost below \$1000 U.S. Impressive, this is.

**Gold Disk**, makers of the popular *PageSetter* and *Professional Page* desktop publishing programs announced a couple of new products. *ComicSetter* does for comics what *ProPage* does for magazines. Each page is composed of rectangular "panels" of any size, allowing color IFF images to be loaded in. Includes a full toolbox of drawing items, and special "Comics" fonts, and support for text-balloons. Full color

output is supported, as are extensive grouping options. Graphic scaling and flipping is also supported, and the user can specify the dots-per-inch (DPI) that the printer will use for the best possible output. It requires 1MB of memory.

GoldDisk also announced *GoldSpell II*, a major upgrade to their original *GoldSpell* program. It works with most word-processors, including WordPerfect, ProWrite, Scribble!, TextCraft Plus, Professional Page, and Page-Setter. It counts the number of words, can grade the level of English in a document, and is lightning fast. Also allows customized dictionaries, and the ability to replace all occurrences of one word with another. For example, US spellings of "color" and "program" can be replaced with "colour" and "programme". Truly a full-featured spelling checker.

*Abacus* is publishing more and more Amiga products. Some new things announced at the show include "Computer Viruses: A High-Tech Disease", a book about the history, definition and direction of viruses. It's 288 pages of protective information, what you can do to prevent and recognize computer viruses. Other Amiga books in their line include "Amiga BASIC" and "Amiga Machine Language". Their software line has expanded to include products like *DataRetrieve*, *BeckerText*, *Assempro*, *TextPro*, to name a few.

*Micro-Systems Software* was displaying their "Word-Perfect Killer" word-processor, *Excellence!*. Excellence provides much of the power of WordPerfect but also includes things like importing

graphics, support for standard Amiga fonts, and so on. Things like macros and mail merge are also all present. The only flaw I can see in Excellence is it's speed. It's not as fast as WordPerfect, but then again, WordPerfect doesn't have to worry about handling multiple fonts. Still, WordPerfect is written in 100% assembly language, and I don't think Excellence is. However, as a word processor, it fares well against it's hefty competitor, and is priced lower. Look for more news on this one!

Not to be outdone, *Word-Perfect Corporation* announced *WordPerfect Library* for the Amiga, which provides a Calendar, File Manager, Calculator, Notebook, and Program Editor all in one, neat package. The version I tried was quite flawless, quick, and efficient. Price unknown.

Perhaps one of the most astonishing things that I saw at the show was a demonstration of the Amiga version of *Dragon's Lair*. Some of you will remember that *Dragon's Lair* was a very popular arcade game, that incorporated Disney-like animation on a laser disk. No computer could ever reproduce these graphics, since it was like actual film. However, if the demo copy of the game is anywhere near what I saw, then some people just may be wrong. By digitizing selected frames from the original videodisk, *ReadySoft* has come up with the most astonishing game I have ever seen. Combining elements from "Gauntlet" and the original "Dragon's Lair", it looks like they've got a winner. Amazingly enough, a spokesperson from ReadySoft told me that the game would

come on 6 disks and still be priced at \$49.95 US. That's an extremely good price. Another interesting note, *Dragon's Lair* is being programmed by a colleague of mine, Randy Linden, who also wrote *The 64 Emulator*.

Some other interesting things I saw at the show include a pair of 3D-glasses. No, not the \$0.50 type of red-blue glasses you can buy at the local convenience store, but true shutter-controlled liquid crystal display glasses. The product was demonstrated with a program that comes with the package, a three dimensional space shoot 'em up. The retail price was \$125 US.

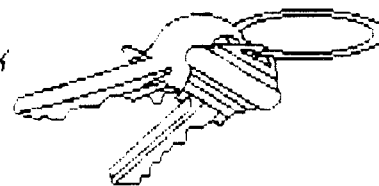
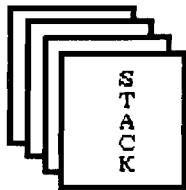
I saw an incredible video product for the Amiga 2000. Called the *FlickerFixer*, this product eliminated screen flicker in interlaced mode from the Amiga totally. Not just a dark screen that goes over your monitor, *Flicker-Fixer* is a product that plus into the video slot on your Amiga 2000. It sends out a true 1/60th of a second full screen image, which requires a multi-sync monitor for that speed. The results are the clearest screens you will ever see. At \$595 US I find this product a little over priced, but not knowing the details of what's involved inside, it's hard for me to say. Businesses and people who do a lot of CAD work will find this gem indispensable. Unfortunately, I did not win the free one that was being given away.

Overall, *AmiExpo* was a great show. It was incredible to see a show that was nothing but Amiga. I highly recommend anyone attending any future shows, if their finances allow, for it is one exciting time!

<BAUG>

# The Amiga

*Keys to Understanding the Stack*



By J. Bos

AmigaBASIC has a command 'CLEAR' which sets the variables to zero and the strings to "". It also takes optionally up to two parameters. The first one changes the default setting of 25000 for the work space, the other one changes the amount of stack.

Some of you may have noticed that the word 'stack' also appears in the 'info' display. This is the display that you get when you click only once on an icon and then select 'info' from the Workbench menu. Let's have a closer look at this so called 'stack'. What is it used for and how does it work? The STACK acts as a scratch pad memory and every high language such as BASIC, Fortran, Pascal etc. uses a STACK. In most cases the stack is used internally, not visible (transparent) to the user. When you write programs in Assembly language or in FORTH, you have to know what a stack is and how it works. As the Amiga is a multi-tasking machine we know that an assembly program may get interrupted several times before it is finished. We also know that it is agreed upon on the Amiga that the processor registers A0, A1, D0 and D1 are considered to be 'scratch' registers. This means that after using them you don't have to go to any trouble to restore the original contents. It also means that those registers may

have been changed when you get them back after an interrupt. For that reason data that is not to be changed without our consent should not be stored into any of these registers. Therefore we may want to use different processor registers, the contents of which first must be saved so that we can restore the contents after we are through using them. If we reserved for storing this information memory in the program itself, we can run into a problem. Because the Amiga is multi-tasking, it is possible that this assembly program is also started on another screen, shell or CLI. If this second program stores its data over top of the data of our first program we will not get the desired results. Also when subroutines are used, the return address must be saved. If another subroutine call occurs before the first one is finished, its return address will be overwritten and lost. Therefore we can not use a fixed memory location to save this return address. Early programmers solved this problem by using a STACK. As an example say that the programmer started his code at location \$200. The first instruction used 3 bytes. Underneath he would write for location \$203 the next instruction of say two bytes. The following instruction would be written on a new line below this for loca-

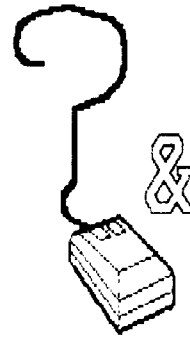
tion \$205. The point to note is that the lowest numbered memory location is at the top of the page and that higher numbered locations are below this. The highest numbered location is at the bottom of the page. After writing many lines of code he might be at location \$3f2. At this point he realized that he needed a stack of approximately 256 bytes. He would skip over memory to location \$500 to continue his code from this location to say \$650. The skipped over memory block would then be treated as a STACK by reserving the first (\$3f3) or the last (\$4ff) memory location as a stack pointer. (This stackpointer could also be located somewhere else in his program. As it became clear that almost every program needed a stack for house keeping chores, all the later processors do have one or more registers dedicated for use as a stack pointer). Lets get back to the assembly program and assume that the first location of hex 3f3 was used as the stack pointer. The next memory location \$3f4 is called the 'top of full stack' because it is on the top of the programmers page and location \$4ff is called the 'bottom of the stack'. The stack pointer is initially loaded with \$4ff (the bottom of the stack). Each time we want to store something on the stack, the stack pointer is decremented before the storing takes

place. This location is called the 'top of the stack'. Note that when the 'top of the stack' coincides with 'top of full stack' it has reached the top of the available stack space.

Adding to the stack is called 'pushing'. The reverse operation is called 'pulling'. The stack at the stack pointer is read first, then the stack pointer is incremented. The stack is empty when the 'top of stack' drops all the way down and coincides with the 'bottom of the stack' location. The assembly program normally checks that the 'top of stack pointer' does not get past the 'top of full stack' or below the 'bottom of the stack' as this would create fatal errors. Two remarks: 1) Some people when listing consecutive memory locations, insist on writing this in the opposite sequence, high numbered locations first. A possible explanation may be in their thinking of associating an 'high' number to belong 'high' on the page. The problem is that in program listings they follow the opposite normal sequence, with the instructions listed in the order of execution from the top of the page down and then complain that the stack is up side down! The bad thing is that many authors of books on assembly language programming have made this mistake and often manage to get themselves confused from chapter to chapter. I have pity with the novice who attempts to understand what such an author is writing about. 2) Stacks on some other micro processors store first when pushing, before the decrement is done. This also works, just make sure that you know what the processor that you are working with is doing.

<BAUG>

## HINTS



## TIPS

I noticed distortion in the graphics display of some programs that previously were all right. For instance the picture of the PacMan program was so badly distorted that the game could not be played anymore. One side of the 'remote control' of DeLuxeVideo looked as if it had been eaten away by rodents. I suspected initially that the disks had developed some bad spots. However the program DiskDoctor did not find anything wrong. I traced it back to the use of 'Preferences'. By using the original 'Preferences' the distortions disappeared. Here is the Hint: If you experience distortions in the display of a program, try to run it with the original 'preference' settings. -J. Bos *Editor's Notes: Some programs rely on you to have your Preferences set to 60 or 80. This is bad programming practice, the programmer should explicitly ask for 60 or 80 columns in the program. To fix the problem, just switch between 60 or 80 columns.*

Most of you have tried by now the CLI and have discovered that it is not hard at all once you get familiar with it. Some of you may have tried to use the commands of the ARP (AmigaDos Replacement Project) and found that those ARP commands are smaller in size, are faster in operation and more powerful. They are also more consistent in the way they operate. It is possible to replace the existing AmigaDos commands with the ARP commands except for two caveats. The first and easy one is that you must have installed the ARP.library file in the LIBS directory. This ARP.library file is supplied together with the ARP commands and documentation on the Fred Fish disk 123 and on the TRANSACTOR disk Vol. 1 issue 2. I have gradually replaced all the AmigaDos commands with the corresponding ARP commands except for the Resident, Copy and Mount commands. I did try the command 'Mount df2:' in my startup-sequence and it crashed the system. So here is the Hint: Suspect the 'Mount' command if you use the ARP commands and the system crashes. I'm using Workbench V1.3 and I have not yet tried to replace the Resident and Copy commands. J.Bos

<BAUG>

## FOR SALE

EXODUS ULTIMA III

\$50

Jason Sarazynski  
759-0727

# 68000 Mnemonics

By J. Bos

68000 Mnemonics	Flags	.BWL	Addressing modes	tsxnzvc	(c)1988 J.Bos	Brantford	Remarks
adda suba		.wl	m0,a		-----		
add sub		.bwl	m0,d	d,m5	xnzvc		
addi subi		.bwl	#,m6		xnzvc		
addq subq		.bwl	8#,m5		xnzvc	8#= 1 thru 8	
addx subx		.bwl	d,d	-(a),-(a)	xnzvc	Z= unchanged if 0	
abcd sbcd		.b	d,d	-(a),-(a)	x??zc	Uses low bytes and x	
nbcd		.b	m6		x??zc	0 - (source+x)	
muls mulu		.w	m1,d		-nz00	32 bit result	
divs divu		.w	m1,d		-nzv0	d/m1 -> 16remain 16quot	
asl/lsl lsr		.bwl	6d,d	3#,d	xnzvc0	--> destination --> cx	
asl/lsl lsr		.w	m7		xnzvc0	--> destination --> cx	
asr		.bwl	6d,d	3#,d	x-z-cn	--> destination --> cx	
asr		.w	m7		x-z-cn	--> destination --> cx	
rol ror		.bwl	6d,d	3#,d	-nz0c	rotate 8, 16 or 32 bits	
rol ror		.w	m7		-nz0c	rotate 8, 16 or 32 bits	
roxl roxr		.bwl	6d,d	3#,d	xnz0c	rotate x + 8, 16 or 32 bit	
roxl roxr		.w	m7		xnz0c	rotate x + 8, 16 or 32 bit	
eor		.bwl	d,m6		-nz00		
and or		.bwl	m1,d	d,m5	-nz00		
andi ori eori		.bwl	#,m6		-nz00		
andi ori eori		.b	16#,ccr		xnzvc	high byte of 16# is 0	
andi ori eori		.w	16#,sr		tsxnzvc	* privileged *	
neg negx		.bwl	m6		xnzvc	For negx z unchanged if 0	
not tst tas		.bwl	m6		-nz00		
clr		.bwl	m6		-0100	read, then clear	
bclr bset		.b l	d,m6	#,m6	--z--	.b=memory .l= register	
bchg btst		.b l	d,m6	#,m6	--z--		
cmp		.bwl	m0,d		-nzvc		
cmpa		.wl	m0,a		-nzvc		
cmpi		.bwl	#,m6		-nzvc		
cmpm		.bwl	(a)+,(a)+		-nzvc	compare memory	
lea		.l	m4,a		-----		
pea		.l	m4		-----	push eff addr	
move		.bwl	m0,m6		-nz00		
move		.w	m1,ccr		xnzvc		
move		.w	m1,sr		tsxnzvc	* privileged *	
move		.w	sr,m6		-----	* privileged 68010+	
move		.l	a,usp	usp,a	-----	* privileged *	
movea		.wl	m0,a		-----		
movem		.wl	m2,Mask	Mask,m3	-----	Mask=d0-d7/a0-a7	
movep		.wl	d,16(a)	16(a),d	-----	peripheral byte move	
moveq		.l	8#,d		-nz00	8# -> 32 bits	
swap		.w	d		-nz00	swap high/low word	
ext		.wl	d		---00		
exg		.l	da,da		-----	data or address	
chk		.w	m1,d		-n???	trap if d<0 or d>source	
link			a,#16		-----	a2s sp2a sp=sp+#16 (neg!)	
unlk			a		-----	a2sp s2a	
Sec.b			m6		-----	set if true, else clear	
DBcc.s			d,16label		-----	Dont Branch if cc or -1	
Bcc bra bsr.s			8label 16label		-----	branch if cc is true	
jmp jsr			m4		-----		
stop			#16		tsxnzvc	* privileged * 16#2sr	
trap			#4		-----	#4= immediate 0 thru 15	
rte					tsxnzvc	* privileged *	
rtr					xnzvc	return + restore ccr	
rts nop illegal trapv					-----	inherent	
reset					-----	* privileged * inherent	

Continued...

DBcc= t f hi ls cc(hs) os(lo) ne eq vc vs pl mi unsigned (for addresses)  
 Bcc= ra sr hi ls cc(hs) os(lo) ne eq vc vs pl mi unsigned (for addresses)  
 DBcc&Bcc gt le ge lt ne eqsigned comparison dest-src

Addressing Modesrelative

	imm	absol.	direct	indirect	+offset	+indexed	pc-offset	+indexed	
m0: #	16	32	d a	(a) (a)+	-(a)	16(a)	8(a,da)	16(pc)	8(pc,da)
m1: #	16	32	d	(a) (a)+	-(a)	16(a)	8(a,da)	16(pc)	8(pc,da)
m2:	16	32		(a) (a)+		16(a)	8(a,da)	16(pc)	8(pc,da)
m3:	16	32		(a) -(a)		16(a)	8(a,da)	16(pc)	8(pc,da)
m4:	16	32		(a)		16(a)	8(a,da)	16(pc)	8(pc,da)
m5:	16	32	d a	(a) (a)+	-(a)	16(a)	8(a,da)		
m6:	16	32	d	(a) (a)+	-(a)	16(a)	8(a,da)		
m7:	16	32		(a) (a)+	-(a)	16(a)	8(a,da)		

Addressing Modes: da= d or a 16(pc)=16label 8(pc,da)=8label(da)

label: or label .,alphanumeric30chars (local labels nnn\$ e.g. 1\$;)

directives rorg offset end equ equr reg set dc dcb ds list nolist spc n  
 page nopage noobj 59<llen<133 ttl 23<plen<101 fail cnop  
 ifeq ifne ifgt ifge iflt ifle ifc ifnc ifd ifnd endc xdef xref  
 macro narg mexit endm include idnt section(code data bss)

```

00000000 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0 Two bytes
           First byte Second byte
00000002 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 One word
           High byte Low byte
00000004 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 One long word
           15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 low byte in 7-0
00000008 Addresses are word aligned (always even)
etc.
    
```

```

|
|
| Sample of a Stack:
00110000 Top of full stack
|
00110014Next word pushed on stack goes here (in 00110014).
00110016 <Top of Stack (Stackpointer points to this location)
00110018 A word pulled from the stack comes from 00110016, the
|stackpointer gets postincremented to point to 00110018.
|
00110100 Top of empty Stack (= Bottom of Stack)
    
```

An address is pushed as a long word (4 bytes).  
 A byte is pushed as a word and located in bits 15-8 !!.

If the stackpointer moves above the top of full stack or below the bottom of the stack it may write over and clobber valid data. This must be prevented by stack checking.

Use of a macro as a subroutine: subnamemacroname a0,d1  
 rts

Vanilla key is Shift-Alt-Control

<BAUG>

