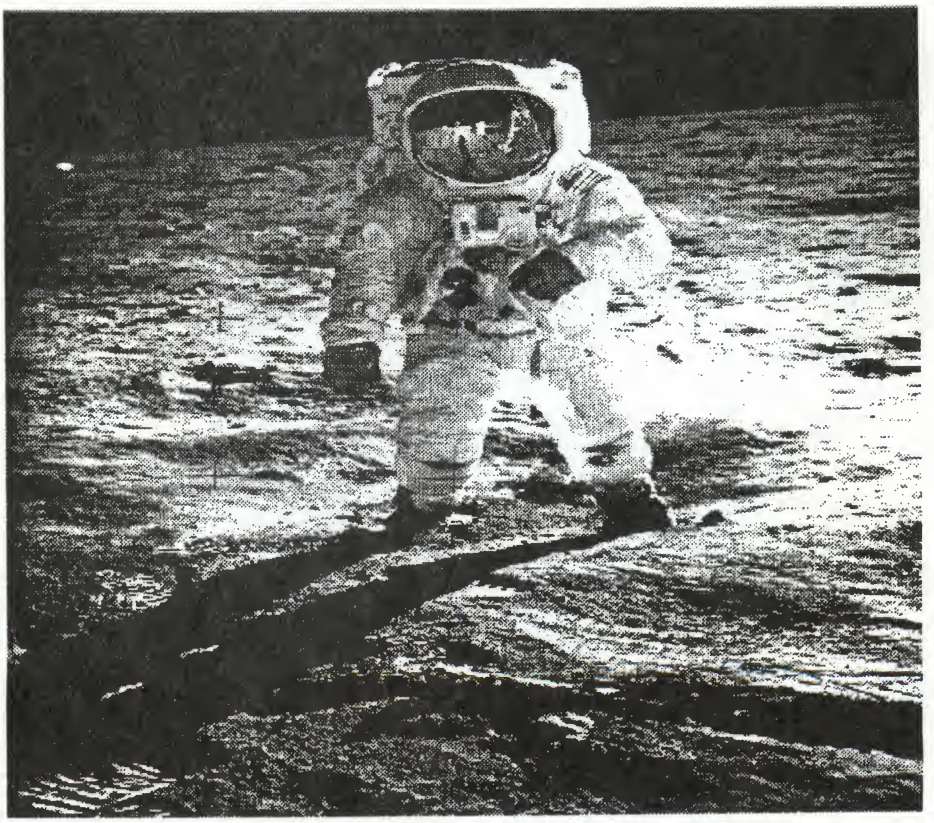
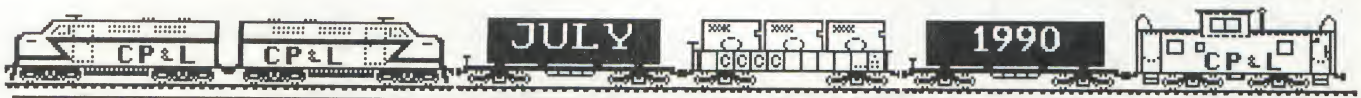
Catalina Commodore Computer Club, Inc.



Vol. 8, No. 7 July 1990

TUCSON, ARIZONA



SUNDAY	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY	SATURDAY
1	2	3 CCCC MEETING 7 PM - 9:30 PM ST. PETER & PAUL GRAMER HALL	4 Independence Day	5	6 FULL MOON	7
8	9	10 EXECUTIVE BOARD MEETING 7:30 PM AT 6150 E. GRANT RD	11	12	13	14
15	16	17	18 Newsletter Party 7 PM St Peter & Paul Madonna Hall	19	20	21 = HELP DAY = 10 AM-2 PM St Peter & Paul Madonna Hall
22	23	24	25	26	27	28
29	30	31				

July Index

COMMUNITY SERVICE

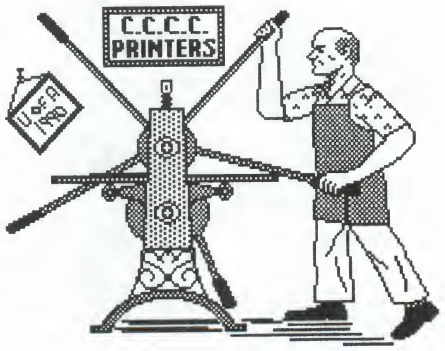
by Leila Joiner, CCCC

I have a habit of reading the Volunteer Center notices in the Tucson newspaper. It finally occurred to me that these often include computer-related items, and that our club would be an appropriate place to look for volunteers to fill these requests. I wrote to the Volunteer Center, offering to publish any computer-related items they received and they sent me a very nice reply, accompanied by their most recent listing.

If you are retired or have extra time on your hands you might want to check our newsletter each month for the Volunteer Center listings. For this month we have the following item:

"DOCTORAL STUDENT welcomes volunteer, 6-8 hours/week: file, organize papers, type (word processor/typewriter helpful), evenings or weekends."

Anyone interested in this, or future, positions should call the Volunteer Center at 327-6207 for further information.



Nybbles & Bytes

by Tom D'Angelo CCCC
296-5076

Over the years that I've been repairing drives and computers, people have asked me how they can determine if their drive is in proper alignment. There are commercial programs available such as "Vorpal" by Epyx and "1541 Physical Exam" by Cardinal. These programs do a good job of screening for bad alignment and will even do a pretty good job of aligning a drive. However, I still believe that using a professional alignment disk with an oscilloscope will do a better job and also allows the technician to check that the disk drive head goes to the same position on a track whether it's moving to that track from the inner or outer part of the disk. This is quite important since it shows "play" or "slop" in the head positioning system.

I would caution users that even though they may be able to "copy" an alignment disk, they will only end up with a non-standardized copy that is of no use for proper alignment. The manufacturer of an alignment disk goes to great pains with special equipment to make sure that the alignment tracks are placed on the disk accurately, all of which is lost when it's copied.

One way a user can roughly check a drive's alignment is by watching the read/write LED on the drive when it is LOADing a program. If there is no copy protection on the program, the LED should come on when the cursor disappears from the screen and stay STEADILY on until the cursor reappears on the screen. If there is any flickering or unsteadiness in the intensity of the LED, it is a possible sign of poor alignment. A good disk to use for this purpose is the Test Demo Disk that comes with all Commodore disk drives. The following is a step-by-step procedure using the Test Demo Disk which can be used to check alignment:

(continued next page)

July Meeting

Bob Clausen will give a demo on how to use Maverick 5. This is a great utility that does everything. COME AND SEE!!!

Ted Sietz will demo the Disk Of The Month!

DON'T MISS IT!!!

(from previous page)

1. To load the directory type LOAD"$",8 then, RETURN.
2. Type LIST then, RETURN. This should list the disk's directory on the screen.
3. Move the cursor to the last program listed placing the cursor at the far left. Then type LOAD and move the cursor to the right of the closing quotes of the file name and type ,8: then, RETURN. Watch the read/write LED for flickering or intensity changes. After a few moments, the LED should go off and the cursor will return to the screen.
4. Cursor to the left side of the screen to the program just above the one you just loaded and repeat the load procedure outlined in #3 above. Again, watch the LED.
5. Keep repeating step #4 until you have loaded all of the programs in the directory. Remember to keep watching the read/write LED for any signs of flickering or intensity variation.
6. Any flickering or changes in LED intensity during the loading of any program suggests that your drive should be checked more thoroughly for proper alignment.

The above procedure is only a rough check of the alignment and will only be as good as the alignment of the disk being used. Drives have the ability to move in or out a half-track looking for data and while they do it, the LED is off. When the LED flickers off and back on, it is indicating that it could not find (in a preset amount of time) the data it was looking for so it switches out of the read mode until it finds the next data it is looking for.

I hope that this article is of some assistance to those of you who are having drive problems and suspect the drive alignment.

Odds & Ends

Bob Clausen is presently streamlining the Club Public Domain Library. Many of the files in the library were greater than 10 years old, which made them obsolete. He is greatly reducing the bulk of the Library, to make it easier for new Library Volunteers

A Portland-based GEOS-SPECIFIC user group, geoMETRIX is publishing a BI-MONTHLY newsletter. GeoJOURNAL subscriptions cost $3.00 per year. Send check or money order to:

RAINBOW SOFTWARE
20224 S. Sprague RD.
Oregon City, OR 97045-9641

CATALINA COMMODORE COMPUTER CLUB
INCOME STATEMENT
For Month Ending May 31, 1990

	END OF MONTH	YEAR TO DATE
REVENUE:		
Newsletter Member Fees		$1,323.00
Newsletter Advertisement Fees	$5.00	20.00
Library Sales Revenue	218.00	517.50
Club Buys	148.00	663.05
Raffles and Auctions	18.00	83.00
Initiation Fees	10.00	180.00
Rental Library Revenue	24.00	24.00
Other Revenue		0.00
TOTAL REVENUE	**$423.00**	**$2,810.55**
EXPENSES:		
Newsletter Expenses	283.55	1,490.70
Library Expenses	16.50	239.50
Rental Library Expenses		0.00
Club Buy Expenses		281.50
Raffle and Auction Expenses		0.00
Meeting Expenses		360.00
Bulletin Board Expenses	39.95	124.28
Membership Expenses		37.50
SIG Expenses		0.00
Saturday Help Day		0.00
Depreciation Expense	9.55	9.55
Administrative Expenses		10.00
Other Misc.		0.00
TOTAL EXPENSES	**$349.55**	**$2,553.03**
PROFIT OR (LOSS)	**$73.45**	**$257.52**

GeoRAM BASIC

downloaded from Q-Link

Ok, so you wanna know how to access the memory in the geoRAM from your own BASIC programs. Well then, here's the skinny, in as simple terms as I can come up with.

GeoRAM, the 512K REU developed by Berkeley Softworks is ONLY compatible with GEOS. Right? Well, maybe...

PAGE

The first thing to understand is what we mean by the word PAGE. For our purposes, a PAGE is any 256 bytes of contiguous memory (that is, any 256 bytes 'in a row'.) For example, memory locations 0 through 255 are the first PAGE in the computer, while locations 256 through 511 are the next PAGE, etc.

The C= 64 contains 256 PAGEs of memory space.

The geoRAM can store up to 2048 PAGEs of data (i.e., 524,288 bytes).

(You might notice that a PAGE is the same size as a 'block' or 'sector' on a disk. Also, since four PAGEs equals one Kilobyte, the 512K of the geoRAM times 4 equals 2048 pages, as stated above. This isn't important to our discussion, but it might help you understand the concept of PAGE.)

Ok... the geoRAM uses 258 bytes of memory space in the computer to help pass information back and forth to computer.

The Page Select Registers

Two of the 258 bytes are called the Page Select Registers. 57342 is the Low Page Select Register and 57343 is the High Page Select Register. (The other 256 bytes are called the geoRAM Window, more about that later.)

Only one PAGE of the 2048 PAGEs in the geoRAM may be accessed at any one time.

To tell the geoRAM which PAGE you want to work on, you simply POKE values into the Page Select Registers. The Low Page Select Register

(continued next page)

(from previous page)
(57342) is POKEd with a value from 0 through 63, while the High Page Select Register (57343) is POKEd with a value from 0 through 31.

Since the Low Page Select Register can be POKEd with 64 different values, and the High Page Select Register can be POKEd with 32 different values, we can access all 2048 PAGEs of the geoRAM. (32*64=2048)

For example, to access the first PAGE in the geoRAM, we POKE 57342,0 and POKE 57343,0.

As soon as values are POKEd to the Page Select Registers, the geoRAM displays its corresponding PAGE in the geoRAM window.

The geoRAM Window

Memory locations 56832 through 57087 in the computer are called the geoRAM Window. To store data in the geoRAM, just POKE the data to these locations. To retrieve it, PEEK the same locations.

When you POKE new values to the Page Select Registers, the geoRAM automatically saves the data from the Window to the proper PAGE of its memory, and displays the data of the new PAGE in the Window.

The word Window is a good description of the memory locations 56832 through 57087, since that memory space in the computer acts like a 'window' through which you can look at the PAGEs of memory in the geoRAM, one at a time.

Example Program

Here is a useless, dirty little program to give you an idea of how the geoRAM may be accessed.

Let's say that we want to fill the geoRAM with keypresses. Don't try to actually use this program, unless you plan to spend many hours pressing 524,288 keys! This is just an example of how the Registers and the Window may be used.

If some of the previous information isn't clear yet, perhaps studying this program will help:

```
 10 HP=57342: REM High Page Select Register
 20 LP=57343: REM Low Page Select Register
 30 X=0: REM starting value for HP
 40 Y=0: REM starting value for LP
 50 POKE HP,X: REM poke value into HP
 60 POKE LP,Y: REM poke value into LP
 70 FOR J=0 TO 255: REM loop 256 times
 80 GET A$: IF A$="" THEN 80: REM get a keypress
 90 K=ASC(A$): REM change keypress to value
100 POKE 56832+J,K: REM poke it to geoRAM Window
110 NEXT J: REM get another keypress and poke it
120 Y=Y+1: REM increment the Low Register
130 IF Y<64 THEN 60: REM get the next page
140 Y=0: REM LP reached 64 so reset to 0...
150 X=X+1: REM ...and increment the High Register
160 IF X<32 THEN 50: REM geoRAM not full yet
170 END: REM geoRAM is full of keypresses
```

(If you wish to actually try the program above, lower the maximum values of X and Y in lines 130 and 160, so that you only store a few thousand keypresses! Then RUN the program and spend some time pressing keys until you get the READY prompt when the program ends.)

To retrieve the keypresses stored by the above program and print them to the screen, we could RUN the following program:

```
 10 HP=57342: LP=57343
 20 X=0
 30 Y=0
 40 POKE HP,X
 50 POKE LP,Y
 60 FOR J=0 TO 255
 70 K=PEEK(56832+J)
 80 PRINT CHR$(K);
 90 NEXT J
100 Y=Y+1
110 IF Y<64 THEN 50
120 Y=0
130 X=X+1
140 IF X<32 THEN 40
150 END
```

(If you try this program, lower the maximum values in lines 110 and 140 to match those you used in the first program!)

By studying the examples above, you should be able to reach an understanding of how the Page Select Registers and the geoRAM Window can be used to access the memory space in the geoRAM. With a little creativity, you might find a use for this in your own BASIC programs.

A Few Notes:

The data need not come from keypresses, it could be read in from a SEQ file on disk, or generated in any number of other ways.

It is not necessary to completely fill every byte in a page. For example, the first byte in each page could be used to hold a value corresponding to the number of bytes stored in that page. (This might be useful for something like storing a database of up to 2048 records of less than 256 bytes each.)

The Page Select Registers cannot be PEEKed, they can only be POKEd. So you must keep track of the last value POKEd. Using a variable (as I did with X and Y in the example programs) is perhaps the easiest way.

If you need any further help with this information, or wish to make any comments or suggestions, feel free to send email to Mink.

MS-DOS COMPUTING

by J.K. Richardson, CCCC

Last month I began a brief discussion of programming languages by presenting a short history of computer programming. This month I would like to continue that thought by proceeding on to some things that should be considered by a programmer (or potential programmer!).

Programming languages are all built on pretty much the same concepts and so resemble each other in more ways than may be immediately obvious. In spite of this, there are some major differences in languages that make choosing the right language for your programming project very important.

There are several classifications of programming languages, so the first task is to understand what it is you want the program (and therefore the computer) to do for you. Here, briefly, are several categories (not arranged in any particular order) of programming languages. Even though the categories are presented here in a rather delineated fashion, there is a degree of overlap between some of the languages.

Scientific languages are generally used for the manipulation of numbers and numeric data. They are designed to be very efficient. Fortran is the best known of the scientific languages.

Systems programming languages are used especially for writing operating systems. Chances are the recreational programmer will not have any reason to program in any of these, but two examples of this are C and Bliss.

Commercial languages are concerned with files of numerical data. This data may be manipulated in various ways and reports printed from it. The best known of these languages is Cobol, which was designed to be used in a business environment. The structure of the language is intended to match the usual method of data flow in a business environment. RPG (Report Generator) is also a fairly well known commercial language. It is (as the name implies) used to create reports, especially when used with other application software.

Interactive languages allow changes and corrections to be made directly from a terminal while a program is executing. These are very convenient for tinkering with the program by allowing the effects of the changes to be seen quickly. The languages LISP and APL are interactive languages. LISP was designed to be used in artificial intelligence, while APL was originally a scientific language. There are certain versions of other languages which are interactive also, such as interactive BASIC. Interactive BASIC should be familiar to most everyone. When you are in GW-BASIC (or some other version) you have probably typed a BASIC statement without a line number and gotten some immediate response. That was the interactive form of the BASIC installed on your computer. Interactive languages such as APL may use more memory and be slower running than compiled languages.

Procedural languages in general have a series of 'statements' which are performed sequentially. Most programming languages fall into this category. LISP is a notable exception, as are PROLOG and RPG.

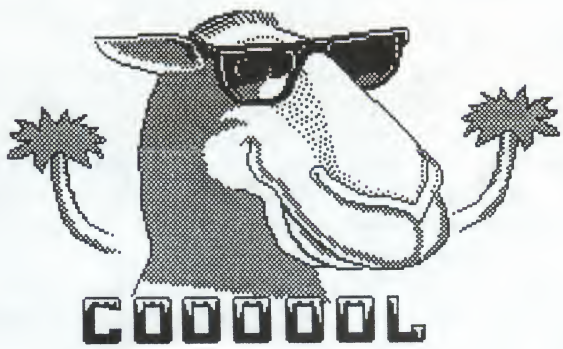
Non-Procedural languages contrast with procedural languages in that you say what it is you want, rather than telling the computer the sequence of steps to do it. For example, you can describe what a certain report is to look like and let the language generate it for you.

There are also a great number of Special Purpose languages which are written for a group having a special need. APT is a language for writing programs to control machine tooling equipment, while SPSS is a statistical program for the Social Sciences. ADA was developed for the military and is based on Pascal. (Ada, by the way, was named for the first computer programmer, Lady Ada Augusta Byron, Countess Lovelace who was the daughter of poet Lord Byron.) Forth is another interesting language, which was developed by C. H. Moore at Kitt Peak Observatory and was first used to control the telescopes there.

Lastly, there is one other category of languages which is of special interest to me. These are languages designed to be used with the computer as a tool for learning. LOGO is probably my favorite.

It is based on LISP, but is designed to be used by children. In spite of this, it has capabilities which allow it to be quite a powerful language. (See me if you want a copy of a PD version.) PILOT is another learning language, and in fact was the first to be dedicated to computer aided instruction. It is an interactive language and is quite easy to learn. It is mainly used for developing programs for teaching.

Next time, I plan to discuss several aspects of choosing a programming language and (hopefully) start comparing a few commonly used (and *interesting*) programming languages.



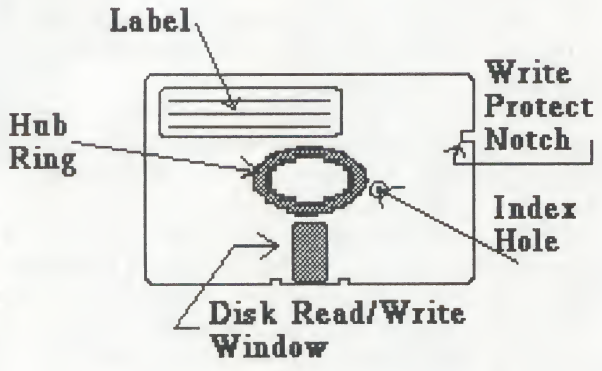
WRITE ON!

by Mike O'Neall, CCCC

SCHOOL'S OUT!! YEAH! Swimming, Disneyland, reading, loafing, the Mall. My kid's kids are full of great thoughts of spending their free time, and what fun it is to listen to their plans.

I have a plan, too. Do you recall the article "KIDS, TOO" from January this year? I was very impressed with the benefits to students of mastering a Word Processor in early school years. Those young ones in Pittsburgh used TWS to write short stories, poems, letters, and most homework with great success. Since I first read about that group I have thought I would like to spend time this summer helping my grandchildren get better acquainted with The Write Stuff. I have been working on an outline of features to be discussed using the same material we have been considering in this TWS beginners column.

My initial session included four kids from 8 to 15 years old. We started at square one, load "BB",8,1, formatted work disks for each, discussed file names, saved, retrieved, designed letterheads, and wrote a couple of paragraphs to someone named George. This included using the extra screen, (CTRL Shift +, CTRL q) loading an



additional file and moving paragraphs from one to the other, clearing the screen a couple of ways, eating and restoring text. I was glad to see that the older ones retained a lot from our few demonstrations over a year ago.

Hoping to stimulate their interest, I started each one on a diary with the encryption command letting each one pick out a secret code. I left them with an assignment to write a short note on letterhead so we have a file to play with next time.

I was delighted to see the typing skills of the eldest, and the enthusiasm of one in the middle. With the youngest it was evident that the computer generation, with no knowledge of a typewriter, needs no explanations of word wrap, but requires a lesson in shifting for capitals. My encouragement came from watching the speed with which each picked up on searching the commands on the back page of the manual and finding page numbers of help in the index. If they can find it, TWS will make it simple to use it. Right? The game plan is to run through the basics, (cursor movements, margins, tabs, italics, underlining, etc.), throw in some fun stuff, (encryption, voices, a mail merge story game), and plan a picnic or party so we can design the invitations. Then, when the need for producing a school paper arrives next semester, hope for the best. In the meanwhile, I have had the opportunity to enjoy their company, and put a small plank in the bridge over the double generation gap.

It would be a pleasure to hear of TWS experiences with young ones in your family, a teaching idea, or a question about a TWS feature. I know it is hard to reach me in person at 887-1969, so I have a message machine and my address is 3740 North Romero Road B17, Tucson AZ 85705.

Latest Discovery: When I learned to type, no one told me to space twice at the end of every sentence. Recently, I have been trying to train my hands to do that, but they are not consistent in their behavior. My writing, including this article, has strange gaps on occasion. If I search (CTRL s) for space space and replace with space, then search for period space and replace with period space space, TWS corrects my erratic spacing without messing with the usual spacing between words. Neat and quick.

WANTED: I am looking for another disk drive to complete a second C64 unit for my grandchildren sessions. Anyone having one for sale, please get in touch.

The C.C.C.C. Meetings are at STS Peter & Paul Catholic Church 4 blocks North of Speedway on Campbell. The General Meeting is the first Tuesday of every month, 7-9 PM, Saturday Help Days occur the third Saturday, 10AM - 2PM

Beyond 512Kb: The Two Megabyte REU

By Andrew E. Mileski
Downloaded from Q-Link

In Volume 9, Issue 6 of the Transactor, Paul Bosacki showed us a miracle; the one megabyte C64. Paul had developed an elegant method of allowing the C64 to access this extra memory out of the C64's own memory map. Unfortunately, this extra RAM is rather difficult to access, not all of it is available to the user, and C128 owners cannot expand their machines in the same manner. The circuit necessary for this feat allows a C64 to use 256K RAM chips, instead of the 64K RAM chips it was designed to use. Although a brilliantly simple circuit, it is over-kill when one already has a **Ram Expansion Unit** (REU) that directly supports 256K RAM chips!

Expanding a REU is relatively simple, and if you already know how to program a REU you can easily take advantage of an expanded one. In fact, all your software that uses a REU is completely compatible! Now our fellow C128 owners can expand their machines easily too, since the extra RAM is accessed out of a REU, and not off a modified motherboard!

In a 512Kb REU, there are two dynamic RAM chip banks on the circuit board. They each consist of eight, 1 bit by 256K dynamic RAM chips, which gives us our 512K bytes. The REU's internal bank register at $DF06 works in 64Kb increments, and has only eight of these REU banks. Banks 0 to 3 are accessed out of chip bank one (U2 to U9), and banks 4 to 7 are accessed out of chip bank two (U10 to U17). Please note the distinction between chip banks, and REU banks.

Dynamic RAM Basics

Dynamic RAM memory chips are constructed with multiplexed address inputs. This means that they carry different information at different times. To access a particular memory cell, we first supply the chip with half of the actual address, a row address. Next we latch this address into the chip by asserting the Row Address Strobe (RAS). Now we supply the chip with the other half of the address, the column address. Once again we latch this address into the chip by asserting the Column Address Strobe (CAS). Depending on the state of the Write Enable (WE) signal, a read or write memory cycle will occur.

Unfortunately dynamic RAMs forget everything unless they are reminded, or refreshed. This must be done about every 4 milliseconds! To refresh the memory a RAS only cycle must be done for every row address. This means that only the row address is latched into the chip; the column address is not needed here. The entire row of data stored in the chip will be refreshed, and our data will be safe for another few milliseconds. In a REU the **RAM Expansion Controller** (REC) chip takes care of all this for us.

Fooling the REC

Since a REU was only designed to access 512Kb of expansion memory, we must fool it in order to access more. We do this buy letting the REU think that it has only 512 kb available at any one time. This means we need some way to switch between the extra banks of memory. The circuit shown in the schematic helps us to do this by adding two bits to the REU bank select register. This two bit output port lets us select one of four banks of 512Kb. So in other words, our REU now thinks it is four separate 512Kb REUs. Now we can easily access up to 2Mb out of a REU!

How the circuit works

The secret to expanding dynamic memory lies in the CAS signal. Since all the RAM chips need to be refreshed with RAS, we don't do anything to this signal and pass it to all memory chips. This leaves us with manipulating CAS.

Manipulation of CAS is the job of IC1, a dual two to four line decoder/demultiplexor. It is used to direct the REC signals CAS0, and CAS1, to the correct bank of eight RAM chips. When one of the REC CAS signals is asserted, the CAS signal of the selected bank is asserted. The bank selection is done with the two select inputs S0 and S1 of IC1.

IC2 and IC3 form a two bit write-only register, whose outputs are the bank select inputs to IC1. Bits 3 and 4 of the REU bank register at $DF06 are latched into IC2, two D type flip-flops, on the negative edge of the system clock (theta2). By mapping our new two bit register to these normally unused bits, the extra memory appears to the user as extra 64K banks beyond the normal maximum of 512Kb. IC3 is a 3 to 8 line decoder/demultiplexor, which is used to decode the lowest three bits of the I/O2 page ($DF00 to $DFFF) address. The IC3 signal O6 is used as the clock signal for the two flip-flops, which latches bits 3 and 4 of the data bus on the positive edge. This happens whenever $DF06 is written to. On a read memory cycle to $DF06, IC3 is disabled and the REU's regular internal register appears on the data bus.

A switch pulls the CLR inputs to each of the flip-flops low when it is closed (position 1). This forces them to select bank zero of 512Kb at all times, which is present in all 512Kb REUs. When the switch is open (position 2), the flip-flops can freely take on the values of bits 3 and 4 of $DF06. This allows complete software compatibility with a 1750 (512Kb) REU.

Lastly, the two Light Emitting Diodes (LEDs), which are optional, simply show us (in binary) what bank of 512k we are using. They allow us to quickly check software compatibility, as explained later.

(continued next page)

(from previous page)

Installation

All the expansion hardware fits inside the REU case. You will not be able to use the RF shield, and it will be a very tight fit with all 2Mb installed (I only went to 1 Mb). Start by expanding your REU to 512Kb, or in other words you need a 1750 REU. See Volume 9, Issue 5 of the Transactor, or the article by ScottB30 on Quantum Link for expanding the 1764. Take the usual static precautions, and of course, any modification to the REU will void the warranty! Proceed at your sole risk!

On the component side of the REU, with the edge connector towards you, locate **Resistor Package** RP3 on the left side near the middle of the board. Flip the board over to the solder side, again with the edge connector towards you. Locate RP3 again on the right side of the board. From the top of the board, find pin 7 of RP3 and cut the trace leading away from the pin. Solder a wire to this pin (pin 7); this is the CAS0 signal. Find pin 3 of RP3 and again cut the trace leading away from the pin. Solder a wire to this pin (pin 3); this is the CAS1 signal.

Flip the board over to the component side with the edge connector towards you, and locate ram chips U2 and U10 on the top left side of the board. Flip the board over to the solder side and again locate these chips. To pin 15 of U2 solder a wire; this is the CASBANK0 signal. Solder a wire to pin 15 of U10; this is the CASBANK1 signal. Run the four wires you now have, down to the right side of the edge connector and secure them in the corner with a piece of electrical tape. This completes all the solder connections to the solder side of the board.

Locate the fifth pin from the right on the edge connector and follow the trace to a component leg; note the placement. Flip the board to the component side for the last time, and locate the component. It is labeled FB2, and should be a **Ferrite Button**, but is a 430 Ohm resistor (yellow, orange, brown, gold bands) on my REU. To the opposite end of the component, away from the edge connector, solder a wire; this is the system clock signal theta2.

Locate the thirteenth pin from the right on the edge connector. Follow the trace to a pass-through, and solder a wire into it; this is the I/O2 signal.

Locate the eighteenth pin from the right of the edge connector, and follow the trace to Ferrite Button FB1. To the side away from the edge connector solder a wire; this is the R/W signal.

Locate the empty pinout (U18) next to the square REC chip. Solder wires into the holes for pins 8, 9, 10, 14, 15, 16, and 28. These are the signals A2, A1, A0, Ground, D3, D4, and +5 volts respectively. This completes the signal hunting.

Now this is where a two inch strip of double-sided foam tape comes in. Lay the strip of tape down the center of the missing IC (U18) pattern you just finished soldering wires into. Try to leave room near the right edge of the case so you can mount a switch. Mount the ICs, in order from left to right, to the tape UPSIDE DOWN (pins sticking up) with the notches (or dots) towards the top of the board. Place them as far apart as you can on the tape. Now connect the ICs using (carefully!) point to point soldering, or wire-wrap DIRECTLY on the IC pins (yes, it can be done). If you do wire-wrap, only 4 wraps per connection are necessary, and "over wrapping" (wrapping on top of wrapping) is recommended for the second connection to a pin. Keep the connections as short as possible eg. +5 volts to pin 14 of IC1 in mine is 1/4 of an inch long.

Mount the switch in the top half and on the edge of the REU's case in a convenient place. I recommend placing it near the bottom right corner, near the plastic post since there is nothing in the way here. A slide switch is neater than a toggle switch, but it isn't very much fun trying to make a square hole for a the slide switch! Mark (or just note) the position of the switch on the case (open: 2Mb, closed: 512Kb) with a marker (Sanford "Sharpie" writes permanently on anything!) or use "Lettra Set" type lettering.

The next step, and last step, is to wire and mount the two optional LED indicators. Solder wires to all the leads of both the LEDs. Drill holes for them at the top edge, of the top half of the case, so they're visible with the REU installed. Make sure to put LED1 on the left and LED0 on the right, so you can read the bank of 512Kb in binary (0=un-lighted, 1=lighted).

Your REU now works the same as before, at least it should! Plug it in, move the switch to position 1 (switch closed, the optional LEDs should both be un-lighted) and give the REU a spin with any test program of your choice. If things look bad, power down and recheck all connections! If it does work, congratulations! You now have an REU that can be expanded to 2Mb by adding a meager (48) RAM chips.

Adding memory

Expand the memory by piggy-backing the existing RAM chips in chip banks one or two, starting with chip bank 1. Bend pin 15 up, to a 45 degree angle, on each of the RAM chips to be added. Connect a wire to this pin before soldering it to the other RAM chip. Once the chip is soldered in place, connect the wire to pin 15 of the next RAM chip to be added. Solder another wire to its pin 15, and solder the chip in place next to the other. Continue in this way until all the RAMs in a chip bank are piggy-backed, and all pins 15 are connected in a daisy-chain fashion in each chip bank. Solder a wire to pin 15 of the right-most RAM chip in the bank; this is the CASBANKx signal. Now use an ohmmeter or continuity tester to see that all the chips in the newly added bank share the same signals on pins 1, 3 to 13, and 15 and 16.

All that is left is to connect the wire CASBANKx to an appropriate numbered signal on IC1. Odd numbered CASBANK signals are for RAM chips

piggy-backed in chip bank one, even numbered ones are for RAM chips piggy-backed in chip bank two. Choose the next available signal when adding a new bank of RAMs. CASBANK0 and CASBANK1 are reserved for the two chip banks already in a 512Kb REU. Connect the CASBANKx signal, and you now have another 256Kb of memory! Repeat for chip bank two for 512Kb extra RAM. Note that you can add as much memory as you like in banks of 256Kb, you don't have to go all the way to 2Mb.

Programming an expanded REU

You don't need to do anything different to use the extra memory in the REU. Just remember that you now have more 64Kb banks than you can access through the bank register at $DF06 when in 2Mb mode (switch in position 2). When in the 1750 emulation (512Kb) mode (switch in position 1), remember you only have access to banks 0 to 7 of 64Kb. Keep in mind the peculiarities of the 2Mb mode as well. That's all there is to it! The number of banks depends on how much memory you added; 0 to 15 with 1Mb, and 0 to 31 with 2Mb for example.

Note that all software that tests for a REU will only find 512Kb available, so a slightly different memory test is necessary to check on how much expansion memory we have. Since the REU can be expanded up to 2Mb in 256Kb increments at the users discretion, a slightly different test is mandatory to figure out just how big the REU is.

Peculiarities

There are a few very minor inconveniences with this expansion project detailed as follows:

1) The REU will not wrap internally between banks of 512Kb, instead the REU will wrap to the beginning of the same 512K bank. For example, saving 2 bytes to $FFFF in bank 7 will put one byte at $FFFF in bank 7, and the other at $0000 in bank 0 NOT bank 8 as expected. This is because the REC chip does not recognize more than 512Kb directly; we have fooled it to use more RAM.
2) Bank register at $DF06 is write-only where bits 3 and 4 are concerned. If $DF06 is read, bits 3 and 4 (as well as bits 5, 6, and 7) will **always** be one no matter what bank of 512Kb the REU is in.
3) Bits 3 and 4 of the bank register at $DF06 are now significant, but in an unexpanded REU they are ignored. This is why we have a switch to disable all but 512K of memory when we run into non-compatible software. See the next section for details.
4) The 512Kb bank select bits are memory mapped by only the three lowest address bits. This means there are images of these bits at $DF0E, $DF16, $DF1E, etc. This is a trivial matter since no sane person uses image addresses!

Compatibility

There is only one source of software incompatibility in this modification; the 2 bit 512Kb bank select register we had added to $DF06. Since these two bits are now significant and didn't used to be, we could have a problem; software that doesn't set these bits to the same value at all times, or sets the bits to a bank of 512Kb that isn't installed yet, won't work. But this is what the switch is for!

The 2Mb modification is 100% compatible with any software written for a 1750 (512Kb) REU, when in the switch selectable 1750 emulation mode (position 1, switch closed). Unfortunately none of the extra memory beyond this can be accessed when in this mode. This is of no consequence, since the software can't make use of any additional memory anyway.

When the 2Mb mode is switch selected (position 2, switch open), we have complete access to how ever much memory we have added. The switch should be in position 2 whenever possible! Most software can function in this mode, including GEOS!

The two optional LEDs are very useful to determine if the software is compatible in the 2Mb mode. If you are using software meant to be used with a 1750 REU, you can use the software in the 2Mb mode if: during a REU transfer the LEDs are always showing the same bank of 512Kb is being accessed (LEDs not flickering), and they indicate a bank that is installed (bank 0 of 512Kb is always present). Without the LEDs, it is simply a matter of saying "It works", or "It doesn't work" when you use software for a 1750 REU.

Final words

Don't let the length of this article discourage you; it only seems difficult in print! Your REU can now contain more memory than an average IBM PC. So be nice to your IBM buddies, and don't brag too much! I can be reached on Q-Link (screen name Recursion), or you can write me directly if you have any questions, comments, or construction ideas. Look for my next article, a much simpler one, on how to add BURST mode capability to a C64; no extra parts required!

NOTE:

Due to the length of this article I was not able to include schematic diagrams or parts list. For complete information regarding the 2Mb REU, call LaserAge BBS at 574-1314 and download the file 'beyond512k.sda'. This self-disolving archive includes nearly all the information that you will need including this article, a schematic and 'tech font' for use with geoWrite. Another support schematic includes a circuit template for the decoder/multiplexor. To further support the 2Mb REU, Berkeley Softworks and Jim Collette created 'configure 2.1' for RAM 1581 and shadowed 1581. All of these files are available from LaserAge BBS, and of course Q-Link.

Independence Day

" A holiday celebrating the adoption of the Declaration of Independence"

The staff at Tony the Printer wish you a *safe* & *happy* holiday

Tony The Printer
3702 E. 22nd St
(S.E. Corner 22nd & Dodge)
(602) 881-8969

AMIGA News

Amiga SIG Meeting

Woods Memorial Library
May 26, 1990

by Macey Taylor, CCCC

Dennis McCormick ran the meeting, and Mike Lawrence showed the DOM, which Joel had prepared. Steve Jess gave the main points of five pages of PageStream press releases (see separate article and DOM) and also said that the US Treasury Department had contracted with CBM for 3000 Amigas over a 4 or 5 year period.

Ken Weaver sent around two copies of the BYTE cover story on the A-3000 series (which we agreed was a little lukewarm) and then discussed at length the June Computer Shopper (see separate article), which sounds like an ad for the A-3000 in many of the articles, even those devoted to other brands. In response to a question from Lynn Doose, Ken said that there are four add-on cards which give the Amiga 24-bit color. Ken also showed us a 44-meg removable cartridge for a Syquest drive. The drive lists for $1195; the cartridges cost about $100 each. He said that optical drives are now available from Sony, Ricoh and two other companies.

Dennis discussed an upgrade problem he had had with Pro-Write, which refused to send him 2.5 for his 2.0 disk, apparently because 3.0 is now out. He wanted 2.5 in order to get the 3.0 upgrade coupon. There is no caveat about time in the 2.5 upgrade info. [It is with incidents like this that a more "organized" user group can help. CCCC is a very well-known u.g. If the club complained, it is likely that the makers of a piece of software would listen.] He also announced that Babbage's (spelling?) in Tucson Mall no longer carried Amiga products. Some day stores will realize that the games market will not support an Amiga store.

I announced that Mike Arrowood, still a member despite his move to Utah, had visited last weekend and reported that his new user group was thriving on our DOM. Mike tried to install ARP on my A-2000 and ran into problems. Apparently the command changes from the floppy I boot from to the WB on the hard disk before installation is complete (or something like this) because the result was a usable but deficient CLI with no icons instead of a fully-functioning CLI with 4 icons behind it (HD, WB, RAM, and RAD). I have a Seagate HD installed on the Bridgeboard. Seeking help -- would like to use ARP.

I passed around two issues of Your Amiga, from the UK (see article) and newsletters with membership forms. I also made a speech (see "Soapbox").

Howard Wooten demonstrated the public domain music editor program MED from Sweden. This seemed a very nice program. It is available on AMI-SW and from our library soon. It has both composition and notation modules. The files can be made into run-time modules. It comes with 30 high quality samples, and it multitasks well.

He also discussed Tiger Cub from Dr. T's, saying it was the best bargain for MIDI with a $99 list and $60+ street price, $15 backup. Nice musical notation printing, sequencer, and works with the Copyist.

Ken Weaver and Mike Van Hoesen demonstrated the Migraph hand scanner and Touch Up software ($395 list). This scanner requires an interface for use with the A-1000. It has no OCR software; everything is saved as an IFF file. It can convert to/from TIFF, DIF, GIF, PCX, DFY, PIC, Atari Degas and is thus compatible with other computer brands. It has only 16 grey scales (Sharp will do 256) and a very l-o-n-g load time. It has a clip function to piece together two scans (all hand scanners need this). User-selected resolutions go up to 400 dpi. Steve Jess said he had a shareware OCR program.

CONGRATULATIONS TO OUR LEADER!!!

Joel Halbert, Amiga SIG leader and co-founder, is the proud father of a new baby boy. Joel was unable to come to the Saturday meeting because of this -- probably teaching his new son how to hold a mouse...

Disk of the Month
May 1990

WB 2.0 - 9 screen dumps of what it looks like.

Utilities - Flashdisk: optimizes floppies; MoveSSP: moves system stack from Chip RAM to Fast RAM; Fast Jet: Desk-Jet graphics speed-up program,

meant for PageStream but may work with Pro Page if you are a hacker.

Pictures - Apollo astronaut scanned shot.

Text - PageStream press releases; DynaCADD, CBM Educational price list as of 5/9/90 (samples: 3000/16-40 & 1950 multisync monitor - $2599; 25-40 & 1950 $3039; 1950 alone $519; 1930 VGA monitor $419. All systems include AmigaVision.); Mice info; Benchmarks comparing 2000 and 3000.

CD August News - Capital District Amiga User Group newsletter on disk.

Code - program by Mike Lawrence to encrypt and decrypt files, with caveats that you must remember your "keyword" because he has no idea how to get the file decrypted without it.

June COMPUTER SHOPPER

by Macey Taylor, CCCC

This issue has a theme of multimedia. The cover shows a Mac IIci, an IBM 55 SX, and an Amiga 3000, with pictures of a music keyboard, a compact disc, a camera, and a "slide."

The Amiga 3000 is reviewed in the "Features" section (but on page 738), as are the other two computers (pages 121 and 196 respectively), by Bob Lindstrom, Editor-in-Chief and noted Apple fan(atic), who discusses at length the "Agnes" chip [sic]. He has, however, done a fair review (fair = just) and addressed numerous items of interest to the "ordinary" user that have not been mentioned in the omnipresent Amiga magazine accounts of the 3000 (though he must not have used Amiga Word Perfect... he calls it a "professional level text processing program"). The article is accompanied by what I consider the most useful and informative visuals yet to appear with a 3000 story.

The real reason to read this issue, however, is the presence of one Amiga rave after another and many implicit boosts for our machine in the non-Amiga articles. You will find such items as big boxes containing "The digitizer only outputs in black and white, even on a color monitor" (price $395, for the Mac) and "Turning a Mac into a full-fledged video processor is like trying to retrofit a car with wings and a jet engine."

In the IBM 55 SX article, there is a section called "Of Mice, Keyboards, and Video." With the 8513 VGA adapter, one gets 704 x 528 in 16 colors, a little slower than the model 70. That's it, folks. The rest is about keyboards, mice, incompatibility problems, noisy drives, and price. This machine lists for $3895 with 30 MB HD, $4295 with 60 MB HD, and it has a maximum expansion to 4 MB RAM.

The Mac IIci (the second most powerful machine in the Mac stable) comes with 1 MB and accepts up to 8 MB RAM, requires many add-ons to achieve its potential capabilities, accommodates only one 3-1/2" internal HD (40/80 MB from Apple; bigger and faster from third parties), has no power switch, includes the required System 6.0.4 (which creates large problems of software incompatibility) and HyperCard 1.2.5, and has manuals that assume Mac expertise. Speeds are given as comparisons to other Macs (no A-MAX...) as 35% faster than a IIx or IIcx similarly configured, 60% faster than a plain II, and 1/2 as fast as the new IIfx. The new Portrait monitor reviewed with it is a 15" full-page monitor with 2-bit monochrome or 4-bit grey scale display. Prices: $8169 list (and another source gives $6000+ as the educational price), plus $1099 for the Portrait monitor, plus $599 for the necessary card to use it, plus $129 for a standard Mac keyboard. The reviewer says that 3 more MB of RAM and a HD are essential.

The multimedia section consists of 63 pages of ads interspersed with text which is divided into 8 articles. "Multimedia" surveys the scene, saying that it began with IBM shipping InfoWindow three years ago, with Apple jumping on the bandwagon a year ago, and that Commodore "has also developed videodisk-controller software." Perhaps some of you remember my showing you a video tape of Microtext in 1988? Amiga Microtext, which has controllers for laser disc players, as well as professional videotape players and home VCRs, has been widely used in Europe and Australia. The Commodore-64 and its near-clone, the BBC Model B (Acorn), have been used for IAV since their introduction. In July, 1988, I participated in a workshop on repurposing laser discs for ESL/literacy work (repurposing is big "business" in my underfunded world). The pitch was made with a little grey Mac hitched to all kinds of stuff to enable it to run a videotape showing a child doing interactive video on a C-64, which needs no extra monitors, etc. since the 1702 is a composite monitor. One of the reasons I bought C-64s (pre-Amiga days, 1983) for my university lab was that I knew many people were using them for IAV... This author did not do much homework.

He discusses the Mac Director software (unfortunate stealing of a name) at length and in combination with MacRecorder for adding digitized speech (see below for more details on these). He partially redeems himself by saying, "Because of its built-in compatibility with ordinary broadcast video, and its relatively low cost, a tremendous amount of video hardware and software is

AMIGA NEWS

available for the Commodore Amiga." He blows it by saying that "Worldwide, about 500 companies and individuals work full-time at the development of interactive multimedia. Their numbers are growing rapidly, but "ordinary" computer users aren't likely to take to multimedia in droves until it becomes just another part of the computer interface." This kind of statement leaves out the whole world of education, where few are able to spend all their time just on IAV, but considerably more than 500 people are spending a large part of their time on it. And all that keeps any Amiga owner from doing IAV is lack of interest.

"Multimedia in the DOS World" is priceless! This author understands his machine. "Lights, computer, action! Well, not quite yet. ...I'm afraid it's going to be a long haul before any budding George Lucas or Steven Spielberg gets anything close to a professional-quality product out of 8086-based PCs.... Unfortunately, going into the opening innings of this exciting new game (multimedia) in the land of 8086 has several strikes against it. First, there can be no question that Commodore's Amiga already has the lead. With its special chips for video and audio work, it is the best-suited microcomputer for multimedia." And he goes on, detailing what is wrong with DOS and OS/2 for multimedia work, plus discussion of the many, many cards needed to do anything. He then describes the "Platform", an 80386, 40 MB HD, with 3 boards, 2 MB video memory and software, totalling $16,000, that one needs to get started, and then says that more memory and a bigger HD are really needed -- to produce 10 frames/sec = jerky output, limited to just a few minutes even with a writable optical drive. Or, you can send your video to Intel; for $250 per minute, they will translate it to a more watchable form. He ends, "...a long time before the average user gets his hands on multimedia power. The technology is at least a generation away for most users."

Skipping "The Pioneering Amiga" and "The Toast is (Finally) Done", we come to "MacFad or MacMarket": "At recent trade shows, IBM, a latecomer to multimedia, and Commodore, an early entry with its Amiga computer, have emphasized the importance of multimedia to counter Apple's presence in the field." This writer details several Mac products. First, the Color Space IIi, your choice of PAL or NTSC video overlay cards (genlock) for $2299.95, or add on to this a Color Space FX card (requires the first as a base) for $3499.95, and you get flicker-free switching between PAL, NTSC, and SECAM, plus RGB conversion. Next, frame grabbers that work at 1 per sec., giving you magnificent 128 x 108, 128 grey scale display, for a cost of only $2599 or $3370 (and that's only the 1MB version; 2MB and 4MB cost an undisclosed amount more). Both require at least a Mac II.

MacRecorder at $249 is a product I know, apparently better than he does. He cites "sampling rates up to 22MHz" and the fact that at 22MHz, 8 seconds of sound eat up 200K. True, but for human speech, the main audio digitizing need in educational and training IAV, samples of 8.5-10MHz are fine. The problem with MacRecorder (the versions I've seen) is that there are only two sampling rates; the other is 7 or 7.5MHz, and that is not adequate for human speech. With Future Sound for the Amiga (at less than half the price, list), you have infinite rates up to 22MHz (I use 9.5 generally), so you can put a great deal more high-quality speech on one disk (plus the extra K on an Amiga-formatted disk). MacRecorder samples at 22MHz do not sound as good as Future Sound samples at 10MHz.

The Director, latest product of its manufacturer, presents 24-bit color at 30 frames/sec, with synchronized digital sound. For only $695. That's assuming you don't want to dump your production to videotape. If you do, cough up another $195 for the Accelerator. Oh, I almost forgot! If you want it to be interactive, gotta buy the Toolkit separately; that's the language that makes it more than just a slide show, and only another $300. These require a Mac+ or higher, with 1MB for B&W, 2MB for color. Unfortunately, after you spend all this money, you'll find it good only for proofing pre-broadcast because of the color restrictions required to run at speed.

However, if you're a serious videophile, you want professional graphics ($699) and 3-D packages ($495-$795), which require 2 floppies (better a hard disk) and 2-4MB RAM.

But it's good to know that, "Right now, much of the multimedia technology is simply too expensive to be accessible to even a serious small business. With the combined emphasis of IBM, Apple, and Commodore, the technology may become competitive and accessible." He ends here. So do I. There are two more less interesting articles in this set.

THE INESCAPABLE TORTURED AMIGA DEVELOPERS SYNDROME

By Edward Anderson

This column contains the opinion of its author and does not reflect the opinions or policies of his employers, the C.C.C.C. or anyone else associated with this newsletter.

In a smoke filled room, lit only by the eerie glow of a single 100 watt light from above, the cast is assembled for the event which was predestined from time immemorial.

Sitting opposed are an Authorized Amiga Dealer and the embodiment of Commodore Corporation itself. Lurking in the background are other pawns in this game controlled from somewhere beyond understanding. The feeling that a larger game is afoot than any human is capable of understanding pervades the air in this seemingly abandoned warehouse in Pennsylvania.

Dealer: OK, Let's thrash this out once and for all!

Commodore: C'mon, we can work this out like adults, can't we?

Dealer: Listen here, bub, we dealers are sick and tired of going 'round and 'round in circles with you about this warranty thing. I thought you were going to do something about these mail order houses.

Commodore: We are doing everything in our power to take care of the problem. These things take time.

Dealer: Look, honoring the warranty makes no sense at all. The warranty card states that there are two kinds of people in this world, Authorized Amiga dealers and End Users. Because, as you know, there are no Authorized Amiga Mail Order houses, that means that the Mail Order houses are the end users and not the people they sell to.

Commodore: Hey! Now wait up there. If we don't honor warranties, we're committing economic suicide. People will be scared to buy Amigas....

Dealer: ...From anyone but Authorized Amiga Dealers. What happened to that "Commitment to Quality, both service and Product," how are you going to guarantee the quality of the machines they sell?

Commodore: That's where you come in. Your quality service will offset that and, besides, the mail order houses are in business, too, and could not stay in business if they sold a whole lot of faulty machines. Then you would have nothing to complain about. I think your argument is about the number of good machines they sell instead.

Dealer: Darn right! And at a hundred to a hundred and fifty dollars less than I can sell them for. These guys have no sales floor, no sales people, no service centers and little overhead. Just a warehouse and a bunch of phones. How can I compete?

Commodore: They also don't have price protection, a line to the parts department or the advertising co-op funds you have.

User1: But the nearest dealer to where I live is over forty miles away. I didn't want to have to make the long haul into town unless I had to.

Dealer: You don't represent the cross section we dealers have a quarrel with.

User2: I didn't see any reason to pay all that money when I could get it shipped to my house and save a bundle.

Dealer: You're the ones we have a gripe with. Don't call me for help with your machine or for hints on Leisure Suit Larry!

Commodore: Now that you admit some mail order purchases are all right, would you honor the warranty on those machines?

Dealer: Well, I sure would be a lot more tempted if you guys would pay me faster for the work I do! It's not money in my cash register you know.

Commodore: It takes a long time to sort through all those NARDA forms, find all the motherboards that dealers have "Stuffed" with bad components so as to swap them out with replacement parts and then get those boards into a saleable form to be shipped back out.

Dealer: Not that I would do that, but I've been told by some less reputable dealers that it saves both you and them a lot of shipping cost and paper work.

Commodore: What, stuffing mother boards or having us repair phoney D.O.A.'s?

Dealer: Where do mailorder houses send their defectives?

Mail Order Dealer: Can I just say something guys?

Commodore and Dealer: BUTT OUT!

Mail Order Dealer: I'm just a business man filling a niche in the market.

Dealer: Yeah, a grey market!!

A blinding light fills the room from a single point of illumination just as some of the occupants are poised to begin pummelling each other. They turn to see the figure of a short, bald, fat man materializing as the light recedes into the glow on the end of the bald man's cigar.

Dealer: Who the heck are you?

Bald Man: I'm the New Tek Fairy Godfather.

Commodore: What are you doing here?

Dealer: Who are you really?

Mail Order Dealer: I don't believe in you.

Bald Man: Don't expect me to do any thing about those Corrupt Disks you left under your pillow then.

Mail Order Dealer: Ok, Ok, I believe!

Bald Man: I am here to suggest to youse guys a way to settle this stink

TWO FER ONE

Game Reviews
by Bernie Joiner

SUBS...

688 ATTACK SUB puts you in charge of the most sophisticated roving arsenal ever developed by mankind. You can represent the U.S. or the Soviet Union through ten different missions. You can hide below thermal layers, launch torpedoes and nuclear missiles, play with the sonar, mess about in the engine room, look at all sorts of display panels, have digitized images of your crew pop onto the screen and speak sub talk, raise and lower the periscope and on and on and on...,but you can't play a friend by modem.

It's a ton of programming crammed into 512K. It's got eye-catching graphics and great sound (except for those sampling glitches). You get to read seventy-two pages of manual!...,but I fell asleep at the helm on the second mission. Since then I've moved onto more stimulating games, and I just haven't managed to get myself to go back and get my dollar value out of...

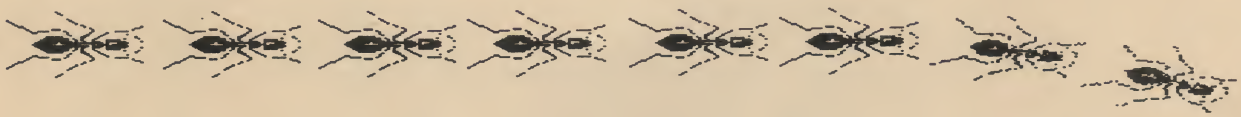
688 ATTACK SUB by Electronic Arts, 512K required, no on-disk copy protection, but you do need the manual to enter code at beginning of game, does not support modem (but it does for the IBM and such), mouse and/or keyboard operated, free Hunter/Killer patch, a ton of manual, $49.95.

TIME...

Mount your bad time thingy and plunge headlong into the mindbending puzzle known as CHRONO QUEST II. Take a weird four-disk ride through Greek mythology, the birth of Christ, the legend of Roland and the Three Musketeers as you attempt to re-knit the fabric of time so you can return to your beloved French chateau in 1922.

The graphics are splendiferous and the opening music absolutely beautiful. Hear ancient people speak. Find lots of neat stuff. Use the stuff. Use the stuff in a precise and exacting order and win the game or lose what's left of your mind. It's great and it's...

CHRONO QUEST II by Psygnosis, mouse driven, requires one MEG, four disks, $49.95.



once and fer all. Now, Commodore's warranty is a non-transferable contract, so whatcha do is charge a nominal Warranty Transfer fee of about a hunnerd to a hunnerd and fifty smackers or there 'bouts according to the items to be repaired for any stuff purchased by mail order.

Mail Order Dealer: Well, we already offer a warranty on our sales for three times the ninety days Commodore offers!

Commodore: We just extended our warranty to one year on purchases of CPUs as of the first of January 1990.

Mail Order Dealer: Oooooops.

Dealer: I would be willing to do that as long as I got a piece of the action.

Commodore: Let's talk this out.....

Now, let me supply you with a glossary of terms used in this little scene for those unfamiliar with the retail lingo.

D.O.A.: A machine which does not function upon receipt. (Dead On Arrival.)

Stuffing: The replacement of good parts in a CPU or other product with defective or broken parts.

NARDA Forms: the lowest form of paperwork known to Warranty techs. This ***MUST*** be filled out for us to receive money from Commodore for any warranty work we do.

Next, some update news. Commodore has been busy working on removing mail order houses from the market and indeed has begun to de-authorize dealers who are doing it. It seems also that some states require warranties to be transferred no matter what is stated on the contract, so that in California, for instance, it is illegal to refuse the warranty on mail order machines, but dealers still have the right to refuse service to anyone. (Especially as long as there is an Authorized Repair Station run by the company itself in the Golden State.)

Commodore is also re-working the pay scale for their service techs, realizing that the fifteen minutes they were paying them to replace a burned out Denise chip covered only the work on the machine and not the paperwork or the customer assistance.

I hope I have shed a little light into the inner view from the "place in between," and that we might all have a little more patience when it comes to the touchier subjects in the business.

There is room in this industry for all sorts (I'm living proof I guess), and there is no reason why Commodore, The Dealers, Third Party Manufacturers and Customers can't be a bit more understanding with one another in their day to day relations.

So the next time your dealer is out of product that they supposedly ordered for you, or you call CATS and they tell you that the 1084P repair kit you ordered is on back order, or some third party device just will not work, remember that the voice on the other end of the phone is a Person also, and they probably have as much control over the situation as you do.

Thank You And Happy Hacking.

Due to the forgetfulness of his instructor, Edward Anderson was left behind here while on a field trip to study Ancient History. The Trans Temporal Timelines Corp. has no office on this planet, so you are all stuck with him.

It can't be too much longer now, so stop complaining!

Two Ways to Destroy Your Disk Drive
(and How to Avoid Them)

by Mike Lawrence, CCCC

Method #1:

The 'CMD' utility that comes with AmigaDOS 1.3 is very handy for redirecting your printer output to a file. Unfortunately it has a bug in it. If you run it from CLI, you are supposed to type something like:

run cmd parallel ram:x opt m n

but if you accidentally type this: *(DON'T TRY THIS!)*

run cmd parallel opt m n

and then use the 'break' command to kill the 'cmd' process, 'cmd' will create a file called 'opt' in the current directory and will continue adding to it's file header until it fills the disk. If you wait until it fills the disk, the file will get closed when you click on the requester that warns you the disk is full. In this case you can just delete the file, and things will be OK (I think).

BUT, if you do what I did and reboot while it is writing all over your disk, (in my case, my hard drive) you may end up with a disk that won't validate, and comes up with one or more 'bad key' errors. You will still be able to read from the drive, but you won't be able to write to it. Fortunately, with the help of a friend I was able to fix the error with the 'disked' utility.

I have reported this bug to Commodore. Hopefully, by the time AmigaDOS 2.0 comes out they will have fixed it.

Method #2:

I had been having a lot of trouble with disks coming up with hard errors in my external floppy drive, so I decided to clean the drive. I had always read that the floppy disk cleaners tended to be abrasive, and could potentially damage your disk heads, so I decided to get a bottle of cleaner and lint-free sponge swabs for cleaning audio/video tape heads and use that.

Although cleaning the drive through the little slot was like building a ship in a bottle, everything seemed OK until I tried to use the drive. As the heads moved around the drive made a clunking sound, and I could see that the upper head was moving up and down when this happened.

It turns out that the upper head in the drive is mounted to a flexible piece of metal, and when I tried to clean the disk heads the swab caught on that piece of metal and bent it. This piece of metal was now catching on something and making the head move up and down.

When I realized what had happened, I opened up the drive and tried to repair it. To get to the damaged part I had to loosen the screws that align the upper head with respect to the lower head, so even though I repaired the damage, the drive now says every disk you put in it is bad or won't validate because the alignment is now messed up.

So to summarize, to avoid these problems:

1. Be careful when using the CMD utility. (Running it from Workbench may be safer than running it from CLI)

2. Don't try to clean your disk head with a cassette tape cleaning kit. Even using a slightly abrasive disk cleaning kit would be better than doing what I did.

!!STOLEN!!

The following property was stolen from your editor's home on June 15.

AMIGA 1000
SN#XM1112186
AMIGA 1080 MONITOR
SN#38391387
STAR NB24-10 PRINTER
SN#300070801898
SUPRA 2400 MODEM
AMIGA 3.5" DRIVE
MEGABOARD 2

Any information, please notify Leila Joiner at 327-0540

Congratulations...

to the incoming Board of Directors...

PLEASE WELCOME

Warren Talbot, President.....574-0732
Frank Traversone, Vice President.....795-3374
Robert Holdcraft, Secretary.....747-3163
Robert Clausen, Treasurer.....747-0196
Tom D'Angelo, Member-at-Large.....296-5076
Frank Prievo, Member-at-Large.....574-0145
Kenney Williamson, Member-at-Large.....750-8164

Amiga SIG Meeting Schedule

Tuesday, June 26 at Harvill Bldg, U of A, 7:30 PM

Saturday, June 30 at Woods Memorial Library, 10:30 AM

Tuesday, July 24 at Harvill Bldg, U of A, 7:30 PM

Saturday July 28 at Woods Memorial Library, 10:30 AM

CCCC, Inc. Newsletter Staff
Leila Joiner, Editor
Warren Talbot, C64/128 Layout
Leila Joiner, Amiga Layout
Columnists:
Mike O'Neall Bernie Joiner
Paul Machula Julia Richardson
Tom D'Angelo Edward Anderson
Calendar by Marv Lossing
Laser Printing by Tom Galloway & Warren Talbot
Please Submit Copy to Leila Joiner
(602) 327-0540

CCCC
TUCSON, ARIZONA

CATALINA COMMODORE COMPUTER CLUB, INC.

NEWSLETTER
VOLUME 8, NUMBER 7 - JULY 1990

**** IMPORTANT ****

* **GENERAL MEETING** - JULY 3, 1990
  St. Peter & Paul Catholic Church
  On Campbell 4 blks N of Speedway
  7 p.m. - Out NLT 9:30 p.m.

* **SATURDAY HELP DAY** - JULY 21, 1990
  St. Peter & Paul Catholic Church
  Madonna Hall
  10 a.m. - 2 p.m.

* **EXECUTIVE BOARD MEETING**
  (All Members Welcome)
  JULY 10, 1990 -- 7:30 p.m.
  Devon Gables Home
  6150 E. Grant Road

MARK YOUR CALENDARS ! !

****

* MEMBERSHIP RENEWAL *
* ADDRESS CHANGE *

Attn., Membership Chairman P.O. Box 32548, Tucson, AZ 85751-2548.

NAME:.....

STREET:.....

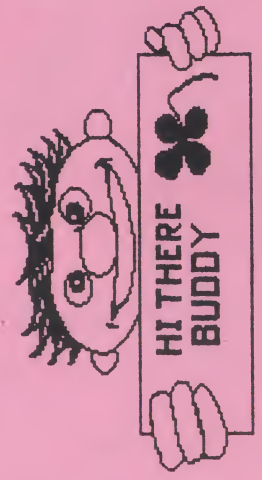
CITY:.....STATE:.....ZIP:.....

PHONE:(.....)..... MEMBER #.....

REMIT CHECK PAYABLE TO CCCC, Inc. FOR $15.00 FOR MEMBERSHIP RENEWAL.
($21 if out of town, $27 if out of country).
New members please also remit $10 Initiation fee.

BULK RATE
U.S. POSTAGE PAID
TUCSON, ARIZONA
PERMIT No. 2567

HELP MAKE YOUR CLUB A WINNER
MEMBER #1069 EXPIRES FEB 1991



Catalina Commodore Computer Club
P.O. Box 32548
TUCSON, ARIZONA 85751-2548

Notify Membership Chairman of any address change. The Post Office does NOT forward Bulk Mail.
CALL FRANK PRIEVO, 574-0145