Volume 4, Number 5
June, 1984

Price
$1.50

CURRENT NOTES

The Newsletter of the Washington D.C., Maryland, and Virginia
ATARI Computer Enthusiasts

800 XL

600 XL

1200 XL

1010

1050

Washington DC Group

NOVATARI: Northern Virginia Atari Users Group

410

AURA: Atari Regional Association of Maryland

810

400

800

850

June 1984

	Sun	Mon	Tue	Wed	Thu	Fri	Sat	
						1	2	
	3	4	5	[6]	7	8	9	<- AURA
NOVATARI ->	[10]	11	12	13	14	15	16	
	17	18	[19]	20	21	22	23	<- DC GROUP
CPM GROUP ->	24	25	[26]	27	28	29	30	

CURRENT NOTES

Current Notes is published monthly (excluding January and August) free of charge by the National Capital Atari Users Group, Inc., 1800 M Street, Washington, D.C., 20036, for its members and for the members of NOVATARI (Northern Virginia Atari User Group) and A.U.R.A. (Atari Regional Association of Maryland). Second-class postage paid at Rockville, Md. Application to mail at second-class postage rates is pending at Rockville, Md. POSTMASTER: Send address changes to Editor, Current Notes, 122 N. Johnson Rd., Sterling, VA. 22170.

The three member groups are independent groups for Atari computer users, and are not affiliated in any way with ATARI, Inc. Opinions expressed in this publication are those of the individual authors and do not necessarily represent or reflect the opinions of any of the User Groups.

Membership dues for each group are $15.00 a year, which includes a subscription to Current Notes. See the last page for further information on club memberships.

Current Notes is available to other Atari User Groups on an exchange basis. Material in this newsletter may be reprinted by other Atari User Groups, provided Current Notes and the author, if applicable, are cited.

Commercial advertising rates: full page, $40; half page, $25; quarter page, $15; 1/6 page $10. Submit photo-ready copy to the editor. The deadline date for both articles and advertisements is the 15th day of the preceeding month.

The Editor of Current Notes is Joe Waters, 122 N. Johnson Road, Sterling, Virginia 22170, (703) 430-1215. News items, short articles, original programs, product reviews, classified ads, and any other material of interest to the membership are eagerly solicited. Classified ads are free to members.

Current Notes Staff

Editor
Joe Waters..... (703) 430-1215

Associate Editor
Staffan Sandberg..... (301) 468-6686

Business Manager
Don Hayden..... (301) 935-2798

Columnists
Jay Gerber..... (703) 525-9715
Bob Kelley..... (301) 839-7377
John Lauer..... (301) 599-1621
Joe Waters..... (703) 430-1215

Contributors
Arthur Corte..... (703) 437-7860
Rochelle Follender... (301) 530-0243
Jim Stevenson..... (703) 378-4093

Editorial:

As you may have noticed, the May issue of Current Notes did not arrive until late in the month. Many circumstances contributed to that delay, but mostly it can be credited to a new editor, new business manager, and new printer -- all in the same month. If you are reading this issue sometime around mid-June, you can rest assured that the experience has paid off and that we are well on the way to establishing a consistent publication schedule. My goal is to eventually have each month's issue arrive on or about the first of the month.

While I'm on the subject of goals, let me mention another one. This issue is exactly 24 pages long, the minimum size we must meet to maintain our second-class mail status. I'd like to publish a longer issue and provide more information to the Atari community. Why don't I? Although I certainly would welcome additional articles from the membership, finding information of interest is not the constraint. I have a good deal of material from Atari as well as from the newsletters of other user groups that I would love to pass on.

The constraint is economics pure and simple. Increasing the publication size from 24 to 28 pages would cost an additional $60 or so. Who will pay for the increased size? Membership dues for the year are fixed and cannot fund any expansion.

Growth can come from three possible sources. An obvious source of income is advertisers. Two half-page ads would just about cover the cost of expanding from 24 to 28 pages. New members, of course, also provide growth. The larger our base, the cheaper it is to publish the newsletter. It cost more to publish 500 copies of a 24-page newsletter than 800 copies of a 28-page newsletter. The third potential source is sales of Current Notes on consignment. Consignment sales not only cover their own cost of production, but also help us take advantage of the economies of scale alluded to above.

So how can you help? You can, of course, help find new members. But more important than that, help me find local retail establishments that would like to advertise in Current Notes or might be interested in carrying the newsletter on consignment. Atari products are not that easy to find anymore. Thus, I feel, local merchant advertisements would benefit our readers as well as the merchant advertising. And, finally, you could try authoring an article or two.

Enough of economics. What's in this month's issue? I'm pleased to introduce a new columnist and also announce the formation of our first SIG (Special Interest Group). John Lauer will provide a monthly column on the activities of the ATR 8000 CPM group (see page 7). Bob Kelly begins an analysis of computer magazines (p.8) and Jay Gerber continues his music tutorial (p.10) and provides a review of Suspended and Ultima III: Exodus. My Basic Beat column departs from its normal format to offer a complete mini database management system (p.12). And from Atari we have features on the joystick ports (p.16), two interesting LOGO programs (p.18), use of the NOTE and POINT commands (p.20), and a simple circle drawing routine (p.21).

Joe Waters

Table of Contents

Index of Advertisers

Washington DC Atari Users Group

1800 G. Street NW, Washington, DC

Club Officers

President.....	Frank Huband	(202) 527-4770
Treasurer.....	Allen H. Lerman	(301) 460-0289
Membership.....	Gerald Whitmore	(301) 459-6164
Program Chairman..	Arthur B. Corte	(703) 437-7860
V Program Chairman	Jim Campbell	(703) 425-1440
Disk Librarian....	Jay Gerber	(703) 525-9715
Tape Librarian....	Bruce Ingalls	(703) 430-3287
ANALOG Disk.....	John Brophy	(703) 425-7169

DC Meetings

are held on the 3rd Tuesday of every month in Room 543 of the National Science Foundation offices, 1800 G Street NW, Washington, DC. The closest subway stop is Farragut West, on the Blue and Orange Lines. Take the 18th Street exit, and walk south (against the flow of traffic) down 18th Street for 3 blocks to G street. The building is on the corner of 18th and G; it can be identified by a sign for the Madison National Bank on the corner. Front entrance is in the middle of the block. Parking is available in the Building, for a fee. The entrance is on the west side of 18th street, between F and G. Meetings begin at 5:30 pm and usually last until 8 or 9.

Coming Events

By Arthur Corte

June 19: In June, providing we can get our communications link working, we hope to demonstrate telecommunications with the Atari and the use of our bulletin board together with CompuServe, the Source and some other timesharing services. Later on in the year, we hope to demonstrate some of the SAT practice programs that have recently become available.

Disk Librarian's Update

By Jay Gerber

There are alot of changes being made in the disk library this month. We now have a disk 'copy house' to do all of our copying for us. This will save wear and tear on my poor single disk drive. This change will allow for an unlimited number of disks to be copied every month. Because of this, I will start handling the ANALOG magazine disks as well as the club diskettes.

This will also mean that we can now sell D.C. library disks to anyone who has the money for them. Starting with the June meeting, I will start selling them at Novatari as well as the D.C. group meetings. The cost of most all of the disks is $4 a piece. I expect to have a complete catalog of all library disks published in the July or August issue of Current Notes.

June will also bring two new disks to the club library. Disk #24 titled 'Storymaker' which was scheduled for release last month, but had to be debugged extensively, will definitely be inducted this month. Also new is an educational disk consisting of programs made by just our members. The topics range from spelling to quantum mechanics, and is numbered #25.

One last item which I don't like or want to announce is the restrictions placed on all future magazine disks. Compute! magazine has indicated to some user groups that its programs are not exactly public-domain. They have set guidelines, and we must follow them. ANALOG has also set rules for its programs and disks. From now on, every person wishing to purchase a Compute or Analog magazine disk, must first show some evidence that you bought the magazine which contains the programs on the disk. This can either be a current mailing label or a copy of the magazine itself. ●

DC/Novatari BBS

BBS Phone Number.....	(703) 425-6698
SYSOP (John Brophy).....	(703) 425-7169
Passwords (Joe Waters)..	(703) 430-1215

Note: Logging on. Call the BBS. After you are connected and answer the linefeed question, the board asks you for name or member number. The next four characters you type must be a pound sign and a 3-digit number, e.g. #264. If you correctly type a member number, you will be asked for your password, andif you type that correctly, the board will respond 'WELCOME, MEMBER!' Your current number and password appear above your name on the label of Current Notes. ●

Novatari: NORTHERN VIRGINIA ATARI USERS GROUP
Greenbriar Community Center -- Chantilly, Va.

Club Officers

President.....	Joe Waters	(703) 430-1215
Vice-President...	Steve Steinberg	(703) 435-2962
Treasurer.....	Curtis Sandler	(703) 734-9533
Secretary.....	Jim Stevenson	(703) 378-4093
Prog Chairman....	* Vacant *	
Disk Librarian...	M. Evan Brooks	(703) 354-4482
Asst. Disk Libr..	Diana Burdt	(703) 425-5073
Membership.....	* Vacant *	

Novatari Meetings

are on the 2nd Sunday of each month in the Greenbriar Community Center on Stringfellow Road in Chantilly, Virginia. Stringfellow Road, also known as Route 645, runs south from US 50 a little more than two miles west of the Fair Oaks Shopping Mall (intersection of I-66 and 50). There is a traffic light where Stringfellow Road meets route 50. The Greenbriar Community Center is on the left-hand side of Stringfellow Road, 1.4 miles south of 50. There is a small parking lot in front and a larger one just north of the center (that is, just before you get to the center). The meeting room is available from 5-9 pm. We offer a BASIC tutorial from 6:00 to 6:40 each month. We also offer a monthly arcade tournament for the young at heart. A short business meeting starts at 6:45 and is followed by brief demonstrations of software. Each month, at least two of the demonstrated programs are given away as door prizes. A formal presentation highlighting a specific software or hardware product begins at about 7:30.

Novatari Minutes

20 May 1984
Jim Stevenson

Announcements: Joe Waters was elected president of NOVATARI thus leaving vacant the position of program chairman. Since many possible programs are already lined up, the job of program chairman should not be overwhelming. It provides an excellent opportunity for new blood to inject new ideas. Also needed is a membership chairman.

Potpourri: A monthly door prize activity was inaugurated. The policy was established that the winner had to be present and a person could win only one prize at any given meeting. A random drawing from the NOVATARI membership list awarded the game Sea Fox to Michael Barry, a new member. A second drawing from the list of those attending the meeting (members or not) yielded the game of Drelbs to Michael Pham. John Baum of STS Video in Falls Church got into the swing of things by contributing spontaneously a diskette storage case as another door prize. Thus a third drawing took place from the attendees list with Roland Gabler being the lucky winner.

Alan Bretzin presented an unannounced demonstration of a "software switchable cartridge" he has developed. He specifically showed how ATARI DOS could be put on a cartridge, and essentially banked-switched in to give full access to all DOS functions without affecting programs still resident in RAM.

Program: Joe Waters demonstrated an ATARI 1027 printer lent by Future Tech of Springfield, Va. Boyd Morse continued the printer review with demonstrations of the Epson and Prowriter. He also showed how the program Megafont could be used to redefine character sets for the printers. ●

Coming Events

June 10: Rene Hertz will demonstrate a four-user system with several ATARI computers sharing common disk drive(s) and a printer. Rene will also demonstrate a software management program that allows one of the units to monitor the activities of the other three.

We will also hold our semi-annual SWAP MEET at the June meeting. If you have any equipment or software you would like to sell, bring it along and see if you can't find an interested buyer. There should be lots of bargains available so don't miss this meeting.

July 8: The 1450 has yet to make an appearance, but that doesn't mean you can't upgrade your ATARI. The ATR 8000 allows you to run CP/M software on your Atari as well as MSDOS. Rob Stewart of Future Tech will tell us all about the ATR and the new capabilities your Atari could have. ●

AURA: Atari Users' Regional Association
Longbranch Public Library -- Takoma Park, Md

Club Officers

President.....	Bruce McLendon	(301) 587-7890
Vice President...	Dave Haseman	(301) 681-5776
Treasurer.....	David Curry	(301) 384-5514
Rec Secretary....	Rochelle Follender	(301) 530-0243
Membership.....	Richard Stoll	(301) 946-8435

AURA Meetings

are held on the 1st Wednesday of every month at 7 pm in Room One of the Long Branch Public Library on Garland Avenue in East Silver Spring. Take the Beltway (I-495) to Exit 29-B South University Blvd East, Route 193). Follow University Blvd. East (Route 193) to the second light (Piney Branch Road). Turn right on Piney Branch Road and continue to the second light (Arliss St). Turn right on Arliss St. past the apartments to Garland Avenue. Turn right on Garland Avenue. The Long Branch Library is on the corner. Park in the Library's lot. Due to constructon, please use the upper-level entrance.

AURA Minutes
2 May 1984

1. Bruce will not be able to attend the meeting tonight or next month. Bob is substituting.

2. COMPUTER AGE is still having a 50%-off ATARI sale.

3. Membership cards are in production.

4. The next meeting is June 6.

5. Silver Spring, Kensington, and Long Branch libraries are possible meeting places for next year.

6. Chris of ACA gave the SYSOP report. It was not a good month for the bulletin board but it should be up soon, between 6 pm and 10 am, only.

7. BASIC XL from OSS gives 5K more bytes.

8. Version C of ATARI BASIC is not yet available.

9. Between 12 and 7 pm Eastern time (with a lunch break from 2-3), the ATARI 800 number is answered.

10. DOS 3 for the 1050 will be available soon.

11. Mo Sherman reported: Review, a free magazine from ATARI is geared to educators. Volunteers are needed to bring equipment. The club's label program doesn't work on the Gemini 10X, but Mo has revised the program and will give it to club members.

12. There is a computer show at the Convention Center next weekend, and one at the Baltimore Convention Center May 10-13.

13. Neal Jacob demonstrated "Auctioneer" a two-player simulation program he wrote and is selling for $11.95 on tape or disk.

14. The library system is being revised. Starting next month, members will receive library cards. New disks (35 and 36) will be available for $4.00. (Checks can be made payable to "AURA".) The archives (all previous disks) can be borrowed from ACA by leaving your library card. For the time being, cassettes will be duplicated on the old, request basis. Annotated disk descriptions will be available on disk.

15. Mike used VISICALC to do his income taxes and he demonstrated his set-up. He thought it was very useful to see the effects changes would make.

16. VISI Corp is filing for bankruptsy. You can buy VISICALC under the ATARI label.

17. Flight simulator II and the Archiver II should be available soon.

18. To get a one-year subscription to "ATARI AGE", send $1 with your name, address, age and telephone number to: ATARI Club Membership Department, 120 Brighton Road, P.O. Box 5065, Clifton, N.J. 07015.

19. ACA has LJK's DATAPERFECT for $49.95.

20. Mo demonstrated EPYX's "Dragonriders of Pern" based on the novels by Anne McCaffrey.

21. If there's enough interest, the club may distribute WABASH disks for $15/10.

Capital ATR Peripheral Micro-Users

By John Lauer

The initial meeting of the ATR-8000 CPM group was held on May 24, 1984. The primary intent of this meeting was organizational in nature. Among the matters discussed were the initiation of a disk library, future location of the meetings, dues, and the relationship to other Atari Users Groups in the Washington, D.C. metropolitan area.

A disk library consisting of approximately 4 to 6 SS/DD disks will be available no later than July. In the development of the public domain software library, initial emphasis will be placed upon program aids/templates covering dBase II, Super Calc, word processing program patches, and utilities designed to improve file handling and disk manipulation. Contact has also been initiated with the Austin, Texas ATR/SIG to obtain several of their program library disks. A nominal fee will be charged to members to purchase all CPM library disks at the meetings to cover operating costs.

The next meeting is scheduled for June 26, 1984 at 6:30 PM (4th Tuesday of month). The meeting will be held at the Public Library in Oxon Hill, Md. The site is located near the Woodrow Wilson Bridge and is just off the beltway. Specfically, the directions are: take beltway to Maryland exit #4 East , St. Barnabas Road(merges to Oxon Hill Rd), proceed 1/4 mile, the library will be on your left . The meeting will be held in the Author Room. (Library phone number is 301-839-2400.)

Since the CPM group is in its formative stages, it would be appreciated if individuals planning to attend the June meeting call Bob Kelly on 301-839-7377 between 6:30-9:00 PM prior to the meeting in order to ensure adequate accommodations are made available at the library.

The feature speaker at the next meeting, Frank Huband, will discuss the ATR Copower-88 hardware option which permits MSDOS operation with the Atari. ●

ATARI SCUTTLEBITS

By Bob Kelly

Over the next two months we are going to examine those computer magazines that clutter our homes. Do we really need to read six magazines or more a month? Is it possible to cut back while still maintaining a growth in computer knowledge? How can we better utilize the time spent reading computer magazines (in other words, get more out of what we read)? Are most magazines really serving the interest of the consumer who pays for the information or just promoting the manufacturers' product? In order to answer these questions and many more, a rating system will be developed and by July you, the reader, should be better able to judge which magazine(s) meets your needs best.

The proliferation of computer magazines continues despite the warnings from Wall Street that a shake-out in the industry is imminent. Presently, there are some 200 to 400 magazines covering the complete spectrum of hardware/software in the market. Not only has the number continued to grow but also the individual size of each magazine. Page proliferation has occurred as more and more ads are placed with individual magazines. The growth in pages per issue satisfies both the need for increased revenues by publishers as well as reader's desire to find "the best buy". I recently overheard a conversation between some postal union employees on this topic. They were discussing union strategy for the next round of contract negotiations. It seems that some union members want to place a limit on the number of pages a computer magazine may have in any single issue. They stated that many magazines are larger than the telephone directory and worst of all require monthly delivery. Alternatively, if this strategy fails, the union will seek a more comprehensive insurance compensation program to cover claims for what has become known as "computer-arm". Could it be that we will be the cause of higher postal rates?

Why do we buy these magazines? The reasons generally put forth include some or all of the following:

--> The desire to find out how to use the machine just purchased (relates to the novice). Needless to say, this is just backwards. Any prospective buyer should read some general articles prior to purchase in order to ensure the computer meets his specific requirements. Sermon aside, the fact remains that many do not read sufficiently prior to purchasing a machine. Thus, it is an important function of a computer magazine to advise the novice. Which magazine do you think is best at this?

--> Users buy the magazines to read the ads for discount prices on hardware/software (not me! I am happy to pay up to 50% more here in Washington, D.C.)

--> Users attempt to improve their programming skills. Magazines supply useful tips and/or sequential monthly lessons. Some magazines have attempted this but the quality as well as clarity of the instructions vary widely. This is an ambitious effort for any magazine and I commend those few that have the "guts" to undertake such efforts.

--> It is a "cheap" way to obtain some games - if you don't mind typing. For us Atari users, it turns out that almost all magazines produce just what we pay for - cheap games. There is a notable exception to this rule. One magazine consistently produces high quality public domain games for the Atari which are worth more than the enforced typing practice. Do you know which one?

--> Those who have owned an Atari for some time realize its potential extends beyond being solely a "game" machine. Thus, they search monthly for practical business or home finance applications, tips, and professionally orientated product reviews. The Atari-directed magazines have so far politely told this group to shove-off (let's publish yet another obscure article on DLI's).

--> Some readers want to obtain useful utility programs designed to improve the efficiency of their computer time. Here many magazines try but few succeed. Those few have produced some great utility programs. Can you name your favorite?

--> Finally, since you are paying $3.00/issue, you expect the magazine to have your interest as its number one priority. Thus, it should keep you abreast of market events affecting Atari so you can make informed judgements regarding the considerable investment you have in hardware/software. Unfortunately for most, the

rule appears to be "let the buyer beware". It often seems that nothing is printed which may reflect adversely upon Atari, or any other computer manufacturer, out of concern that they may not advertise in their magazine. In my opinion, there is one exception - although its editorials sometimes miss the point.

Hold it ... we have just created our rating system. Let's review it again.

Criteria for Rating Atari-Related Magazines:

1. Instructional content for novice.
2. Ad content diversity.
3. Programming tutorials.
4. Game quality.
5. Professional uses, product reviews.
6. Utility program development.
7. Consumer awareness.

Now, let's look at some of the magazines against which we will apply our criteria.

The top five micro-computer magazines, ranked by circulation, plus the circulation figures for the Atari-only or Atari-related magazines are as follows:

Monthly Circulation Data
(as of January 1984)

1.	Personal Computing	528,000
2.	Byte	428,000
3. *	Compute	420,000
4.	Popular Computing	347,000
5. *	Creative Computing	305,000
	* Personal Software	180,000
	* Antic	100,000+
	* Analog	80,000
	* Atari Connection (Quarterly)	160,000

Sources - Wall Street Journal, April 9, 1984 and direct contact with publishers.

* Denotes those magazines that will be examined in this column. Note, of the 160,000 in circulation for the Atari Connection, about half represents subscription and newsstand sales.

It is immediately obvious from the above list that solely game review magazines are not included - the reasons why I will leave to individual interpretation. However, I will add (I can't resist), this is one area where individual subscription costs could be cut quite easily. Further, one, possibly two, Atari related computer magazines that should qualify for the list are not included. The reasons for their exclusion relate to a number of factors. Suffice it to say that the intent of this article is to cover the most popular and widely read computer magazines dealing with Atari. Finally, some of the magazines are not designed to meet all the criteria by which we will rate them. An initial impression may be that this will yield an unbalanced rating system. On the other hand, I believe that if a magazine is to intentionally narrow its scope, it should be clearly superior in those categories it does cover if it is to warrant your purchase.

Well, next month, just like the old movie serials, we will find out who did what and to whom. See you then ..... Oh, by the way, comments and suggestions on your favorite magazines are welcome prior to July 15, 1984. Send to: Bob Kelly, 8309 Bella Vista Terrace, Ft. Washington, Md. 20744. The most interesting comments will be published in the August issue of Current Notes.

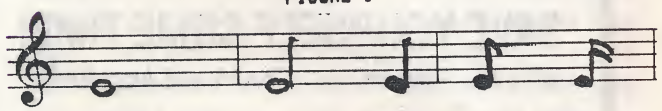
Musical Notes

By Jay Gerber

Hello, and welcome back to Musical Notes. Last month we were discussing what sound is and how your Atari makes it. This month I will explain how to make different lengths of specific frequencies (see last month's column) into a logical pattern favorable to listen to, which is a good definition of the word music. The best way to learn about anything is to have something tangible in front of you, instead of trying to comprehend a mass of theories. So I have drawn diagrams and will try to explain them.

Sound, as I stated in the last column, is a set of vibrations played at a speed between 20 cycles and 20,000 cycles per second. If you play this frequency for a specified amount of time, you produce what is called a note. There are different types of notes, each based upon a different set amount of time. Figure 1 has a set of the most used durations in music, and their corresponding note values. The first one is called a whole note, and is the basis for the rest of the notes. A whole note is approximately four seconds long. Every other note is a fractional part of this and each one has a duration of the specified fraction of a whole note.

FIGURE 1



Whole Half Quarter Eigth Sixteenth

For instance, a quarter note (third from left) has a duration of about one second. Going back to the BASIC statement: SOUND 0,121,10,10 from last month, we can write a small BASIC program to play any of these note values. Figure 2 is such a program. When RUN, the program will ask you for a note value. Type 'W' for a whole note, 'H' for a half note, 'Q' for a quarter note, and so on. The program will play a frequency for that specified amount of time.

The number 121 I said, was the particular frequency that corresponded with the note called 'middle C'. Well, you may ask, just what is this mysterious middle C?! Technically, each frequency value that is put in the SOUND statement is a musical note. However, they are all not 'true notes' because some of them do not fit in the chromatic scale. The chromatic scale is a set of pitches that define all the notes playable by an instrument such as a piano.

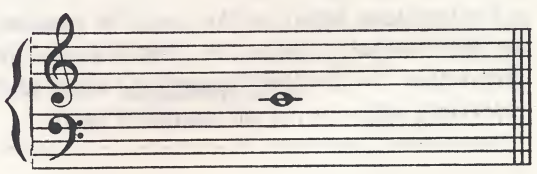
FIGURE 2

```
10 DIM A$(1),N$(5),T(5)
20 N$="WHQES"
30 FOR I=1 TO 5:READ X:T(I)=X:NEXT I
40 DATA 4,2,1,.5,.25
50 GRAPHICS 0
60 POSITION 6,3:? "Type of Note   Duration (Sec)"
65 POSITION 6,4:? "============   =============="
70 POSITION 8,5:? "WHOLE":POSITION 28,5:? "4"
75 POSITION 8,6:? "HALF":POSITION 28,6:? "2"
80 POSITION 8,7:? "QUARTER":POSITION 28,7:? "1"
85 POSITION 8,8:? "EIGHTH":POSITION 27,8:? "1/2"
90 POSITION 8,9:? "SIXTEENTH":POSITION 27,9:? "1/4"
95 POSITION 6,12:? "TYPE OF NOTE (WHQES)";:INPUT A$
100 TIME=0
105 FOR I=1 TO 5:IF A$=N$(I,I) THEN TIME=T(I)
106 NEXT I
110 IF TIME=0 THEN GOTO 95
120 FOR I=18 TO 20:POKE I,0:NEXT I
130 SOUND 0,121,10,10
135 SEC=(PEEK(18)*65536+PEEK(19)*256+PEEK(20))/60
140 IF SEC<TIME THEN GOTO 135
150 SOUND 0,0,0,0
160 GOTO 95
```

Music can be translated from hearing sounds directly on paper. The note durations in figure 1 are the symbols used for varying lengths of musical notes. They are placed upon a set of lines called a staff, and their position on the staff represents the musical note value.

Figure 3 is the musical staff. Each line and space between lines represents a different frequency. The funny curly-cue symbol taking up the upper five lines of the staff is called a treble clef. Similarily, the backwards 'c' with two dots next to it on the bottom five lines is called a bass clef.

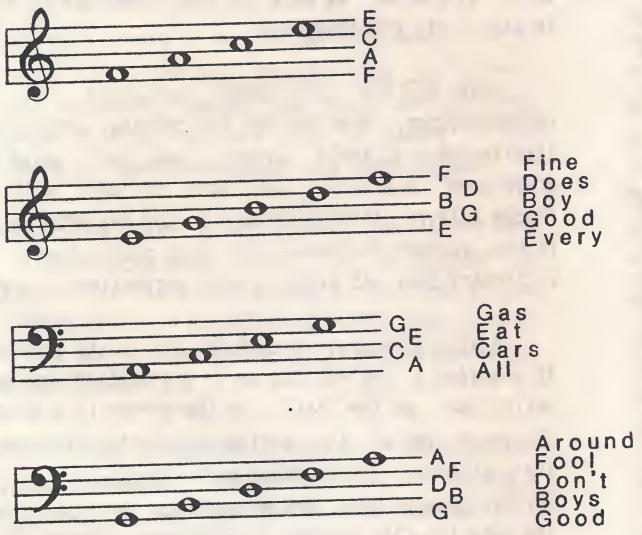
FIGURE 3



The bass clef is a subdivision of notes that are typically low in frequency. In order for a note to be low, and in the domain of the bass clef, it has to be in a position under middle C. And, in order for a note to be called high and in the domain of the treble clef, it must be above middle C. So, middle C must be the divider between the low and high notes.

If we go to the space between the line that middle C is on and the very next line above it, we are at a frequency called D. Looking at figure 4, the next note, which is the line above D, is called, strangely enough, E. F is the next note, which happens to reside on the space between the first (E) and second (G, as you might have already guessed) lines above middle C. Now comes the tricky part. The very next space is not H. There are no more musical note names above G. So, what is that position called? It is called A, because the notes just go in circles. You can start at any note you want, then go alphabetically until you reach G, then you go back to A and start over again (and again and again...)

FIGURE 4



You might also notice another problem. There are more than one of every note. How can that be, you might think. In between the third and forth lines there lies a note labeled C. This is 7 notes up from middle C. We give both of these notes the same name, C, but we say they are in different 'octaves'. An octave is all of the notes between and including two notes that have different pitches yet the same name. Two notes in different octaves sound similiar, except one is lower or higher. Since the exact relation in sound between a note in two different octaves is very hard to describe in words, I will try to show you through the use of BASIC.

Type in a new variation of the ever-popular statement: SOUND 0,121,10,5 This will reduce the sound to half the volume. Now type in SOUND 1,243,10,5. This is telling the computer to produce a pure sound out of channel 1 at a frequency somewhat less then middle C (121). Listen carefully to the two notes being played. They should sound the same, yet one is much lower than the other. They are both the same note, C, but are in different octaves. If you go down 7 notes from middle C, you are at the note represented in BASIC as the value 243.

So, now you have a 'map' of the musical staff. One hint in trying to remember what note names go where is to memorize little phrases or words, where each line or each space is represented as a letter in the phrase. Looking at figure 4, we notice that if we start at the first space of the top staff and go up by spaces, the note names spell the word 'FACE'. A well-known phrase for learning the lines on the top staff is 'Every Good Boy Does Fine', where the first letter of every word represents the note name of each line. Starting with the bottom line of the Bass staff and working up line by line, we get 'Good Boys Don't Fool Around', which is a logical conclusion to the aforementioned phrase. The spaces on the Bass clef go with the phrase 'All Cows Eat Grass' or, for modern times, 'All Cars Eat Gas'.

Alot of you are probably saying: Hey, what about sharps and flats, knucklehead! Well, since I am running out of room, I will discuss them next month along with using the note names and length values to play actual music.

Please send contest entries (see last month's column), complaints, quips, and comments to:

Jay Gerber
3639 N. 36th Road
Arlington, Va. 22207

BASIC BEAT

By Joe Waters

For the past four month s we have been working on a diskette library program as a means of illustrating various BASIC techniques. This month we were to start work on actually building our database. We will build a database, but it won't be exactly what you anticipated.

As part of the standard NOVATARI meetings, I teach a simple BASIC tutorial. For the first several months of the tutorial, we worked on an arcade-type program called BLOCKADE. With that finished it was time to start work on another program. I offered several different possibilities and the one selected was a program called WORD BUILDER.

This program can be used as a vocabulary builder. It offers two basic options. In one, a word appears on the screen along with five alternative definitions. By moving the joystick, the player moves from one definition to another. Pressing the button indicates the definition selected. If the selection is correct, a new word and five new definitions appear. If incorrect, the definition selected disappears and the player gets to choose again. The program is timed and each player tries to get as high a score possible in the time allotted. The second game option is essentially similar except that the player sees one definition and five possible words to match it with.

In the original version of WORD BUILDER, I included 100 different word/definition combinations in data statements within the program. However, in preparing the program for the tutorial, it occurred to me that this format was inappropriate. Not only would it be uninformative to list a program with 100 DATA statements, but the words I was using might not be appropriate for many other users. It would be far better if each user could construct his or her own set of word/definition combinations.

The WORD BUILDER game can be used by children to improve their vocabulary as well as by adults. However, the basic data needed by the program to make it interesting to an adult would make it far too difficult for a second-grader. Similarly, if you have several children of different ages, you might want different vocabularies for each child. Thus, it would be nice if the same program could use any of several different databases.

With these considerations in mind, I decided to provide a DICTIONARY BUILDING program as a front-end to the WORD BUILDER program. Using the DICTIONARY program, each user could build unique, individual, dictionaries that would then be used as the basic data file used by the WORD BUILDER program.

In constructing the WORD BUILDER DICTIONARY program, I found that it had all of the basic features of a database management system. I decided, therefore, to depart temporarily from our previous project, and devote this month's column to this simple utility program. (I confess that I was also heavily influenced by the time constraints imposed by creating this program and also assuming the editorship of Current Notes.)

First of all, what do we want to accomplish? We want a program that will allow us to build and maintain one or more databases of words and their corresponding definitions. Build and maintain means we should be able to add records to the database, revise information already in the database, and delete records we no longer want. Of course, we want the user interface to be easy to use, fast, and foolproof.

WORD BUILDER DICTIONARY satisfies all these considerations. When you run the program, after a brief display of a title screen, you are asked which grade-level dictionary you want to work with. You choose a level (from 1 through 9) and the program reads in the current dictionary for that grade level. If a dictionary does not exist, one is automatically created.

A single display is used for all update procedures. It provides a box for the word and another box for the definition. At the bottom of the screen is a prompting to remind you of the various one-letter commands that are available: First, Last, Next, Previous, Quit, Add, Delete, Change, Sort, and Write. The firt four commands are used to step through the database. Typing "F" (for First) positions you at the beginning of the dictionary. The first word in the dictionary and its definition automaticlly appear in the appropriate boxes. Typing "N" (for Next) brings up the second word. "L" (for Last) takes you to the end of the dictionary. "P" (for Previous) displays the previous word in the dictionary. Of course, safeguards are included to prevent you from stepping beyond the last (or before the first) word.

To add a word to the dictionary, enter "A". The area inside the WORD box is whited out and you simply type in the new word. Constraints are built into the program so that only reasonable characters can be input into either the word or definition. In particular, none of the control characters will work and you cannot move outside of the box except by pressing the RETURN key. When you have typed in the word and pressed RETURN, the program first checks t see if that word already exists in the dictionary. If it does, the current definition for that word is automatically displayed in the definition box. If not, the cursor is moved to the definition box and you can input the definition you want. When you are done with the definition, press RETURN and the word is added to the end of your list of words and definitions.

Since the Add function automatically searches the current word list to see if the word is already in the list, it can be used as a locate command. That is, if you want to find any particular word in the list, use the Add option and the program will first search for your word and then display it when, and if, it finds it.

To delete a word, enter "D". The program will print a questioning prompt to be sure you really want the word and its definition deleted. If you answer Yes, the word is deleted.

If you see a word or definition you would like to change, enter "C". You will be positioned in the word box. Enter whatever changes you want, or if none, simply press RETURN. You are then positioned in the definition box. Make your changes and press RETURN.

Words do not have to be sorted, but if you want them in alphabetical order, enter "S" to trigger the sort option. This is a simple bubble sort and, if you have a hundred words in the dictionary, can take several minutes. If you have a faster sort routine available, go ahead and use it.

None of these changes will have any impact on the database stored on your diskette until you use the Write option. When you do, the entire list currently in memory is written out to disk. When you choose the Quit option, you will be reminded of the necessity of saving your changes on disk and given an opportunity to do so if you like.

Although this program was written to build dictionaries for WORD BUILDER, it can be used to make any number of alternative databases. For example, what if, instead of words and definitions, you stored a state and a state capital or a year and the best picture for that year or a jersey number and one of the Redskins? Any of these could be used as inputs to the WORD BUILDER program (which I will publish in this column some time in the future) to create a quiz game for states, movies, or football.

The complete program is listed below. It assumes a word size of at most 10 characters and a definition size of 36 characters. Variables are dimensioned to hold up to 100 words. You could, of course, make the word size larger and allow for more than 100 words by changing the appropriate lines. I will discuss how the various parts of this program fit together in the BASIC tutorial at the NOVATARI meetings over the next few months.

```
10 REM ************************************************
15 REM *          WORD BUILDER DICTIONARY             *
20 REM *          By Joe Waters, (c) 1984             *
25 REM ************************************************
30 REM This program allows you to build one or more
35 REM dictionaries for use with WORD BUILDER. Each
37 REM dictionary can be geared to a different grade
40 REM school level. Words can be added, deleted or
45 REM revised as needed.
50 REM ************************************************
52 REM
55 GOSUB 855:REM Write out title, initialize, get data
60 GOSUB 415:REM Update dictionary as needed
65 END
66 REM
70 REM ************************************************
75 REM *                  SHORT SUBROUTINES
80 REM ************************************************
84 REM
85 REM GCHAR: Get single character from keyboard
90 OPEN #2,4,0,"K:":GET #2,X:X$=CHR$(X):CLOSE #2:RETURN
95 REM ************************************************
99 REM
100 REM GLINE: Get line of input from keyboard reading
105 REM upto LIMIT characters starting at pos. XI,YI.
110 J=0:REM J keeps track of character count
112 POSITION XI+J,YI+2:? CHR$(20):REM Print pointer
115 GOSUB GCHAR
120 IF X=155 THEN POSITION XI+J,YI+2:? " ":GOTO 165
125 IF X=126 AND J=0 THEN GOTO 140
130 IF X=126 THEN X$=" ":GOTO 145:REM back space pressed
132 IF J=LIMIT THEN GOTO 140
135 IF X>=32 AND X<=95 THEN J=J+1:GOTO 145
140 FOR V=14 TO 0 STEP -1:SOUND 0,180,10,V:NEXT V:GOTO 112
145 POSITION XI+J-1,YI+2:? " ";:REM Erase pointer
146 POSITION XI+J-1,YI:? X$;:REM show character on screen
150 IF X=126 THEN POSITION XI+J,YI+2:? " ";:J=J-1
160 GOTO 112:REM Get another character
165 M$="ONE MOMENT...":GOSUB MESSAGE:M$=""
170 REM Transfer screen contents to M$ array
175 FOR I=1 TO LIMIT:LOCATE XI+I-1,YI,Z
180 IF Z=160 THEN POP :GOTO 190
185 M$(I,I)=CHR$(Z):NEXT I:RETURN
190 M$(I)=BLANK$(I,LIMIT):RETURN
```

```
195 REM ************************************************
199 REM
200 REM LINE: Draw line of length LL
205 FOR I=1 TO LL:? CHR$(18);:NEXT I:RETURN
210 REM ************************************************
214 REM
215 REM BOX: Draw box length=LL+2, top left corner=X,Y
220 POSITION X,Y:? CHR$(17);:GOSUB LINE:? CHR$(5);
225 POSITION X,Y+1:? CHR$(124);
226 POSITION X+LL+1,Y+1:? CHR$(124);
230 POSITION X,Y+2:? CHR$(26);:GOSUB LINE:? CHR$(3);
232 RETURN
235 REM ************************************************
240 REM MESSAGE: Display messageS centered on line 17
245 POSITION 0,17:? BLANK$(1,38);
247 XL=(40-LEN(M$))/2:POSITION XL,17:? M$;
250 FOR J=14 TO 0 STEP -0.8:SOUND 0,100,10,J:NEXT J
255 SOUND 0,0,0,0:RETURN
260 REM ************************************************
264 REM
265 REM PAUSE: Wait awhile
270 FOR JJ=1 TO WAIT:NEXT JJ:RETURN
275 REM ************************************************
280 REM WDATA: Write dictionary out to disk file FN$
285 M$="WRITING DICTIONARY TO DISK ...":GOSUB MESSAGE
290 CLOSE #1:OPEN #1,8,0,FN$
295 PRINT #1;TOTAL
300 FOR I=1 TO TOTAL:GOSUB GWORD
305 PRINT #1;WDEF$
310 NEXT I:CLOSE #1
315 M$="FILE SAVED; ENTER NEW COMMAND":GOSUB MESSAGE
316 RETURN
320 REM ************************************************
324 REM
325 REM SDATA: Sort words alphabetically
330 M$="SORTING LIST ...":GOSUB MESSAGE
335 FOR I=1 TO TOTAL-1:POSITION 30,17:? I;
340 FOR J=I+1 TO TOTAL
342 I1=46*(I-1)+1:I2=I1+9:J1=46*(J-1)+1:J2=J1+9
345 IF WORD$(J1,J2)>WORD$(I1,I2) THEN GOTO 360
350 I3=I1+45:J3=J1+45:WDEF$(1,46)=WORD$(I1,I3)
355 WORD$(I1,I3)=WORD$(J1,J3):WORD$(J1,J3)=WDEF$
360 NEXT J:NEXT I
365 M$="LIST SORTED; ENTER NEXT COMMAND":GOSUB MESSAGE
366 RETURN
369 REM
370 REM ************************************************
375 REM PWORD: Put word/def into WORD$ array
380 I1=46*(I-1)+1:I2=I1+45
385 WORD$(I1,I2)=WDEF$(1,46):RETURN
389 REM
390 REM ************************************************
395 REM GWORD: Get word/def from WORD$ array
400 I1=46*(I-1)+1:I2=I1+45
405 WDEF$(1,46)=WORD$(I1,I2):RETURN
410 REM ************************************************
415 REM *                Display menu of choices
420 REM ************************************************
425 GRAPHICS 0:POKE 752,1
430 POSITION 8,1:? "WORD BUILDER DICTIONARY";
435 POSITION 12,3:? "Total Words: ";TOTAL
440 POSITION 2,20:LL=36:GOSUB LINE
445 POSITION 2,21:? " First  Last  Next  Previous  Quit";
450 POSITION 2,22:? " Add  Delete  Change  Sort  Write ";
455 POSITION 2,23:GOSUB LINE
460 POSITION WX-1,WY-2:? "Word:";
462 LL=10:X=WX-1:Y=WY-1:GOSUB BOX
465 POSITION DX-1,DY-2:? "Definition:";
467 LL=36:X=DX-1:Y=DY-1:GOSUB BOX
470 IF NUMBER>0 THEN GOSUB DDATA
474 REM
475 REM ************************************************
480 REM *                Get Individual Commands
485 REM ************************************************
490 POSITION 25,3:? TOTAL;"  ";
495 POSITION 2,19:? "ENTER COMMAND:      ";
500 IF TOTAL=0 THEN NUMBER=1:X$="A":GOTO 520
505 GOSUB GCHAR:FOR I=1 TO 10
510 IF X$=ANS$(I,I) THEN POP :GOTO 520
515 NEXT I:M$="(INVALID COMMAND!)":GOSUB MESSAGE:GOTO 490
520 POSITION 17,19:? X$
525 IF X$="F" THEN NUMBER=1:GOSUB DDATA:GOTO 490
530 IF X$="L" THEN NUMBER=TOTAL:GOSUB DDATA:GOTO 490
535 IF X$="N" THEN NUMBER=NUMBER+1:GOSUB DDATA:GOTO 490
540 IF X$="P" THEN NUMBER=NUMBER-1:GOSUB DDATA:GOTO 490
545 IF X$="A" THEN GOSUB ADATA:GOTO 490
550 IF X$="W" THEN GOSUB WDATA:GOTO 490
555 IF X$="S" THEN GOSUB SDATA:GOTO 490
560 IF X$="C" THEN GOSUB CDATA:GOTO 490
565 IF X$="Q" THEN GOSUB EXIT
569 REM
570 REM If none of above, must be delete option
575 M$="* DELETE: ARE YOU SURE (Y/N) *":GOSUB MESSAGE
580 GOSUB GCHAR:IF X$<>"Y" THEN GOTO 595
585 WORD$(46*(NUMBER-1)+1)=WORD$(46*NUMBER+1)
590 TOTAL=TOTAL-1:NUMBER=NUMBER-1
595 GOSUB DDATA:GOTO 490
600 REM ************************************************
604 REM
605 REM DDATA: Get word/def and display on screen
610 IF NUMBER<>0 THEN GOTO 615
612 NUMBER=1:M$="(FIRST WORD IN LIST!)":GOSUB MESSAGE
615 IF NUMBER<=TOTAL THEN GOTO 620
617 NUMBER=TOTAL:M$="(LAST WORD IN LIST!)":GOSUB MESSAGE
620 I=NUMBER:GOSUB GWORD
625 POSITION WX,WY:? WDEF$(1,10)
626 POSITION DX,DY:? WDEF$(11,46)
630 M$="WORD NO. ":M$(10)=STR$(NUMBER):GOSUB MESSAGE
631 RETURN
```

```
635 REM ************************************************
639 REM
640 REM ADATA: Add new word to dictionary
645 IF TOTAL<100 THEN GOTO 650
647 M$="DICTIONARY FULL!":GOSUB MESSAGE:GOSUB DDATA:RETURN
650 POSITION DX,DY:? BLANK$(1,36)
652 POSITION WX,WY:? WHITE$(1,10)
655 M$="(INPUT NEW WORD)":GOSUB MESSAGE
660 XI=WX:YI=WY:LIMIT=10:GOSUB GLINE:LET NEWW$=M$
665 IF TOTAL=0 THEN GOTO 700
670 REM Check to see if WORD is already in dictionary
675 FOR I=1 TO TOTAL:GOSUB GWORD
679 REM
680 IF WDEF$(1,10)=NEWW$ THEN POP :GOTO 690
685 NEXT I:GOTO 700
690 M$="(WORD ALREADY IN DICTIONARY!)":GOSUB MESSAGE
695 NUMBER=I:GOSUB GWORD:GOSUB DDATA:RETURN
698 REM
699 REM Fill box with WHITE spaces
700 POSITION DX,DY:? WHITE$(1,36)
705 M$="(INPUT DEFINITION)":GOSUB MESSAGE
710 XI=DX:YI=DY:LIMIT=36:GOSUB GLINE:LET NEWD$=M$
715 TOTAL=TOTAL+1:I=TOTAL
717 WDEF$(1,10)=NEWW$:WDEF$(11,46)=NEWD$
720 GOSUB PWORD:NUMBER=I:GOSUB DDATA:RETURN
725 REM ************************************************
729 REM
730 REM CDATA: Change word/definition
735 M$="Revise Word":GOSUB MESSAGE
740 POSITION WX,WY:? WDEF$(1,10):POSITION DX,DY:? WDEF$(11,
745 XI=WX:YI=WY:LIMIT=10:GOSUB GLINE:LET NEWW$=M$
750 M$="Revise Definition":GOSUB MESSAGE
755 XI=DX:YI=DY:LIMIT=36:GOSUB GLINE:LET NEWD$=M$
760 WDEF$=NEWW$:WDEF$(11)=NEWD$:I=NUMBER:GOSUB PWORD
765 M$="Word Revised; Enter New Command":GOSUB MESSAGE
766 RETURN
770 REM ************************************************
774 REM
775 REM EXIT: Leave program
780 GRAPHICS 0
785 ? :? "NOTE: IF YOU HAVE NOT USED THE WRITE"
790 ? "OPTION, NO PERMANENT CHANGES HAVE"
795 ? "BEEN MADE TO THE DICTIONARY DATABASE. "
800 ? :? "WOULD YOU LIKE TO WRITE THE CURRENT"
805 ? "VERSION OF THE DICTIONARY TO THE "
810 ? "DISK BEFORE EXITING?"
815 POSITION 2,10:? "PLEASE ANSWER Y OR N : ";
820 GOSUB GCHAR
825 IF X$="Y" THEN ? X$:GOSUB WDATA:END
830 IF X$="N" THEN ? X$:END
835 SOUND 0,150,12,10:WAIT=20:GOSUB PAUSE
836 SOUND 0,0,0,0:GOTO 815
840 REM ************************************************
845 REM * Display Title, Initialize Variables,       *
846 REM * and Read in dictionary from disk.          *
850 REM ************************************************
855 GRAPHICS 2:SETCOLOR 2,0,0:POKE 752,1
860 POSITION 8,1:? #6;"DATA"
865 POSITION 5,2:? #6;"DICTIONARY"
870 POSITION 8,4:? #6;"for"
875 POSITION 3,6:? #6;"word  builder"
880 ? "          By Joe Waters"
885 ? "            (c)  1984"
886 REM ************************************************
889 REM
890 REM Program constants
895 GCHAR=90:GLINE=110:LINE=205:BOX=220:MESSAGE=245
900 PAUSE=270:PWORD=380:GWORD=400:EXIT=780
905 WDATA=285:SDATA=330:DDATA=610:ADATA=645:CDATA=735
910 WX=14:WY=8:REM The x,y location of the word
915 DX=2:DY=13:REM The x,y location of the definition
920 NUMBER=0:REM Position of WORD in list
924 REM ************************************************
925 TOTAL=0:REM Total no. words currently in dictionary
929 REM
930 REM Dimension arrays
935 REM Program assumes max word size=10 and max
940 REM def size=36. (Combined word/def size of 46.)
945 REM Variables dimensioned to hold upto 100 words.
949 REM
950 DIM NEWW$(10),NEWD$(36):REM Used for new word/def
955 DIM WDEF$(36+10):REM WDEF$ holds word/def combination
960 DIM WORD$(101*46):REM Holds list of word & defs
965 DIM BLANK$(46):REM Used to erase screen
970 DIM M$(38):REM Used to write messages
975 DIM WHITE$(38):REM Used to print white spaces
980 DIM ANS$(10):REM An array of acceptable answers
985 DIM X$(1):REM Used for one-character inputs
990 DIM FN$(15):REM Filename of dictionary file
991 REM ************************************************
994 REM
995 REM Initialize arrays
1000 BLANK$(1)=" ":BLANK$(46)=" ":BLANK$(2)=BLANK$(1)
1005 REM Enter inverse video space for initializing WHITE$
1010 WHITE$(1)=" ":WHITE$(38)=" ":WHITE$(2)=WHITE$(1)
1015 WORD$(1)=" ":WORD$(4600)=" ":WORD$(2)=WORD$(1)
1020 ANS$="FNPADSCQWL"
1025 WAIT=200:GOSUB PAUSE
1026 REM ************************************************
1029 REM
1030 REM Determine grade level of dictionary file
1035 GRAPHICS 0
1040 POSITION 2,2:? "ENTER GRADE LEVEL FOR DATA FILE ";
1042 INPUT X$
1045 IF X$>="1" AND X$<="9" THEN GOTO 1060
1050 M$="(ENTER A NUMBER BETWEEN 1 AND 9)":GOSUB MESSAGE
1052 GOTO 1040
```

```
1054 REM
1055 REM Construct name for data file
1060 FN$="D:DICTIONA.RY":FN$(14)=X$
1065 GRAPHICS 0
1070 ? :? "WORDS FOR GRADE LEVEL ";:? X$;" ARE"
1075 ? "STORED IN ";:? FN$(3,10);FN$(12,14)
1079 REM
1080 REM Determine if file exists
1085 TRAP 1090:OPEN #1,4,0,FN$
1090 CLOSE #1:STATUS #1,Z
1095 IF Z=1 THEN TRAP 40000:GOTO 1120
1100 IF Z<>170 THEN ? "ERROR NO. ";:? Z:STOP :REM ERROR?
1101 REM ************************************************
1104 REM
1105 REM Create file; store total words(0) 1st record
1110 ? :? "NEW FILE.  INITIALIZING DICTIONARY..."
1115 OPEN #1,8,0,FN$:PRINT #1;TOTAL:CLOSE #1:RETURN
1119 REM
1120 REM Read in data from dictionary
1125 OPEN #1,4,0,FN$:INPUT #1;TOTAL
1130 ? :? "DICTIONARY CURRENTLY HAS ";:? TOTAL;:? " WORDS."
1135 IF TOTAL=0 THEN GOTO 1170
1139 REM
1140 REM Read in each word/def combination
1145 ? :? "READING IN DICTIONARY ... "
1150 FOR I=1 TO TOTAL:INPUT #1;WDEF$:GOSUB PWORD:NEXT I
1155 NUMBER=TOTAL:L=LEN(WORD$):IF L=4646 THEN GOTO 1170
1160 REM Fill rest of WORD$ with blanks
1165 WORD$(LEN(WORD$)+1)=BLANK$:GOTO 1155
1170 RETURN
```

ATARI Portability

ATARI, Inc.

The four controller jacks on the 400 and 800 are connected to the PIA's 2 eight bit bi-directional ports. This allows the jacks to be used for output as well as input. Because these ports are easily made TTL (transistor transistor logic) compatible, the designer of peripherals will find them easier to use for some things (like simple sensors and controllers, etc.) than the serial port. The following program and explanation shows how to program and use these ports.

```
100 PORTA=54016: REM Port A Data & Direction Registers
110 PORTB=54017: REM Port B Data & Direction Registers
120 PACTL=54018: REM Port A Control Register
130 PBCTL=54019: REM Port B Control Register
140 ALLOUTPUT=255:SETDIRECTION=56:SETDATA=60
150 POKE PACTL,SETDIRECTION
160 POKE PORTA,ALLOUTPUT
170 POKE PACTL,SETDATA
180 POKE PBCTL,SETDIRECTION
190 POKE PORTB,ALLOUTPUT
200 POKE PBCTL,SETDATA
210 FOR D=0 TO 255
220 POKE PACTL,DPOKE PBCTL,D
250 NEXT D: END
```

(Continued on Page 22)

Nibbles & Bits

By Jay Gerber

This month's N&B will focus on two different types of current adventure games. Suspended, by Infocom, is an all-prose (just words) format. ULTIMA III: EXODUS, by Origin Systems, is a graphic role-playing game. Both games are relatively new and typical of both types of computer adventures on the market today.

Suspended is an inter-active text adventure game for one player. Like its predecessor the Zork trilogy, it will understand whole sentences that you type in. In Suspended, you are in a state of suspended animation, which means you can't move, but your brain can think. Your thoughts are translated by a computer that runs an entire world. It just so happens that a massive earthquake has just hit the complex where the computer that controls the environment, transportation, and food supply for your world are housed. Since you can't move, you use six robots that move around the complex for you to adjust and hopefully restore the computer to working condition.

Each of the six robots you control with your thoughts has different attributes and weaknesses. Waldo, for instance, has six extensions and can manipulate almost anything. However, he is very sensitive, and it can be quite frustrating if he won't fully co-operate. Since each robot's forte is a different sense (like sight, hearing, touch, etc.), you must use all your robots together to accomplish the quite challenging task.

In Suspended, as in all Infocom games, you must use solely your wits to sucessfully win the game. You will very often be given a complex problem which you must solve before you can do anything else. While this makes the game very frustrating indeed, you really feel elated when you solve the challenges. The description of a certain room will vary depending on which robot is perceiving it. For instance, Auda the audio droid, will only tell you what she hears. Sensa, the sensory robot, can tell you only what vibrations emanate from the room. And Poet, who is a diagnostic drone, does see, but as his name implies, his descriptions can be very oblique and confusing.

The documentation in Suspended is excellent. A plastic-coated cardboard map of the complex is included with six tokens that represent each of the robots and help with the troublesome task of keeping track of the robots' current locations. The manual is loaded with background material and explains the game very well.

I found the game easy to learn, since you talk in English sentences and the computer responds with incredibly detailed descriptions. The puzzle complexity is extremely tough, so those adventurers who are looking for quick solutions should look somewhere far away. The vivid scenes of Suspended's novel-like plot are great. I felt like I was actually in that world with the robots next to me. I loved this game even though I still haven't solved it!

Ultima III:Exodus is a breakthrough in adventuring in many ways. Most obvious is that more than one person can adventure at a time. Ultima III can handle up to four characters of different classes. This means that you can have a party with wizards, fighters, theives, etc., and they all, to a certain extent, can be independently controlled. Another revolutionary feature is the fantastic sound track that accompanies the action. Mood-setting music in four-part harmony makes the game seem like more of a movie. It gets dark and foreboding when monsters appear and solemn when you approach the king.

Your mission is to traverse the beautiful scrolling landscape in search of the evil demon Exodus, and then, using all the strength and courage you can muster, defeat the powerful creature. Exodus is esentially a role playing game (RPG). An RPG works this way: you create a hero using point values assigned to different attributes such as strength, dexterity, wisdom, and intelligence. You have 50 points to divide among these categories. The attribute you put the most points in determines your character class. For example, a fighter needs strength and dexterity. A wizard needs intelligence, and so on. You then choose your sex and race (human, elf, dwarf, etc).

With your created character you then do various things to find clues and gain experience and strength in order to defeat Exodus. You can purchase supplies in numerous cities, fight monsters in dungeons and on land, sail in ships, even drink in bars.

In terms of puzzle-solving difficulty, Ultima III:Exodus is not hard at all. You mainly gain clues and the strength you need through time (lots of it!). The combat phases are very nicely animated and realistic. The dungeon scenes are in an impressive 3-D perspective. The greatness of this game lies in the detail. The landscape and cities are ornately detailed. Your interaction with the other characters of the land is incredibly detailed. Although your actions are pretty much limited to several one-letter commands, I didn't feel at all constrained in making my actions. Ultima III is far superior to the first two Ultima adventures which have very similiar gameplay.

LOGO Lessons

From ATARI Demopac #11

USING SETREAD AND SETWRITE
More Than Just Dribbling
JG/6/83

The Atari Logo Reference Manual states that SETREAD and SETWRITE may be used to create a "dribble" file by recording the screen output to either a printer, disk or cassette. There is another use for SETWRITE that is not mentioned in the ATARI manual. SETREAD and SETWRITE function like the "OPEN" command in ATARI BASIC. They open an Input/Output Control Block (IOCB) for reading or writing to a device (casette, disk or printer).

The Atari Logo Reference Manual describes the SAVE command for saving the contents of the workspace to a specified device. The drawback to using this method when saving the workspace is that the procedures are saved in the order that they were written in, not in the logical order in which the are used. This can make for hard-to-read programs. The following program uses the procedure PRINTER to print readable listings to the printer:

```
TO PRINTER
SETWRITE "P:
PR [TITLE]  PR []
PR [NAME AND DATE] PR []
PO [PROCEDURE NAMES ... ]
SETWRITE []
END
```

PRINTER opens a channel to the printer with the SETWRITE "P: command. The next line prints any title for the heading of the listing. PR [] prints a blank line for formatting purposes. Then the name and date are printed. Another blank line and then the command PO [PROCEDURE NAMES ...] prints out the procedure names in the order listed inside the brackets. This allows control over the sequence of the procedures in the printout. If three procedures "ONE, TWO, and THREE" were created in the order THREE, TWO and ONE and they were prited to the printer using the SAVE "P:, they would be printed in the sequence THREE TWO ONE. But if they were printed using the PRINTER procedure, PO [ONE TWO THREE], they would be listed in the sequence ONE TWO THREE. The SETWRITE [] command is used to close the file and return output back to the screen. It is important to use the SETWRITE and SETREAD commands within a procedure to avoid the "dribble" effect mentioned above.

SETWRITE can also be used when saving procedures to disk. If you have ten procedures in your workspace and you want to save three of them to a disk file, SETWRITE can be quite useful. Instead of erasing all the procedures you don't want and then using the SAVE command to save the three you want the following procedure can be used:

```
TO SAVE.DISK
SETWRITE "D:DEMO
PO [ONE TWO THREE]
SETWRITE []
END
```

The file DEMO now contains the desired three procedures. Files created with the above procedure can be LOADed just like a file saved using the SAVE command. It is not necessary to use SETREAD to read files saved in the above manner.

VIDEO TURTLE
The Joystick and the Turtle
JG/9/83

VIDEO TURTLE is a program that teaches the turtle to "listen" to the joystick. The turtle will respond to the commands from the joystick to set its heading to any increment of 45 degrees and then take five steps forward. Pressing the joystick trigger will change the turtle and pen to the next of three preset colors and all following commands will then be carried out in the current turtle and pen color until the trigger is pressed again. (The preset pen and turtle colors in this version are red, yellow and blue.)

The pen and turtle colors are changed by using a WHEN demon in the SETUP procedure. The WHEN demon enables the Atari Logo collision detection and sets up a demon to constantly check for a collision or event. In this case the event is the pressing of the joysick trigger, one of the 20 types of collisions or events that WHEN demons can detect.

To make this program usable to younger turtle fans and to give more variety to the turtle's activities, a HELP menu is included when the program is started. The HELP menu can be removed any time a full screen graphics display is desired. While the basic draw and turn commands are given with the joystick, the HELP menu suggests other commands that may be given through keyboard entry. The DRAW procedure contains the KEYP predicate that checks for a keypress and executes the desired intruction via the PEN.UP.DOWN procedure.

```
TO START
INSTRUCTIONS
SETUP
HELP
DRAW 5
END

TO INSTRUCTIONS
SETBG 1
TS CT SETCURSOR [10 2]
PR [VIDEO TURTLE]
PR [] PR []
PR [PLUG IN JOYSTICK IN PORT #1]
PR []
PR [TO  CHANGE THE TURTLE'S  PENCOLOR PRESS THE JOYSTICK
BUTTON]
PR []
PR [THE TURTLE STARTS WITH THE YELLOW PEN DOWN] PR [] PR
[]
PR [TO RECALL COMMANDS PRESS H FOR HELP]
SETCURSOR [4 20]
PR [* PRESS ANY KEY TO BEGIN *]
KEYPRESS RC
END

TO DRAW :STEP
IF KEYP [PEN.UP.DOWN RC]
CHECKJOY JOY 0
DRAW :STEP
END

TO KEYPRESS :ANYKEY
END

TO CHANGEPN
IF PN = 2 [SETPN 0 STOP]
SETPN ( PN + 1 )
END

TO CHANGETURT.CLR
IF PN = 0 [SETC 15]
IF PN = 1 [SETC 75]
IF PN = 2 [SETC 35]
END

TO SETUP
CT CS FS ST TELL 0
SETPN 0
SETPC 1 75
SETPC 2 35
SETBG 1
SETC 15
WHEN 3 [CHANGEPN CHANGETURT.CLR]
END

TO HELP
CT SS
PR [U PICKS UP THE PEN]
PR [D PUTS DOWN THE PEN]
PR [X PUTS REVERSING PEN DOWN]
PR [C CLEARS SCREEN]
PR [E ERASE HELP]
END

TO PEN.UP.DOWN :UPDOWN
IF :UPDOWN = "E [CT FS]
IF :UPDOWN = "H [HELP]
IF :UPDOWN = "X [PX]
IF :UPDOWN = "C [SETUP]
IF :UPDOWN = "U [PU]
IF :UPDOWN = "D [PD]
END

TO CHECKJOY :POS
IF :POS < 0 [STOP]
SETH 45 * :POS
FD :STEP
END
```

The NOTE and POINT Commands

Atari Inc.

[Ed: The following program, distributed by ATARI Customer Relations, illustrates the use of the NOTE and POINT commands to perform random access to data files. I have added REM statements throughout the program to help explain the various sections.]

```
1 REM RANDOM ACCESS - ATARI BASIC
2 REM BY KENT SMITH
3 REM MODIFIED BY JOE WATERS (MAY, 1984)
8 REM Set total records, record length, and
9 REM length of name and address
10 RCDS=100:LNGTH=40:NM=20:AD=20
19 PRINT CHR$(125)
20 DIM S(RCDS),C(RCDS),SPACE$(LNGTH),NAME$(NM),ADDR$(AD)
24 REM Clear out the strings
25 SPACE$=" ":SPACE$(LNGTH)=" ":SPACE$(2)=SPACE$
27 NAME$=SPACE$(1,NM):ADDR$=SPACE$(1,AD)
28 REM If index file does not exist, error
29 REM occurs.  Jump to line 4000.
30 TRAP 4000:PRINT CHR$(125);
35 POSITION 7,3:PRINT "READING INDEX FILE"
39 REM Open both files to make sure they exist.
40 OPEN #2,4,0,"D:INDEX.DAT"
50 OPEN #1,4,0,"D:FILE.DAT":CLOSE #1:TRAP 40000
59 REM Reopen data file for UPDATE
60 OPEN #1,12,0,"D:FILE.DAT"
69 REM get the index to the data file.
70 FOR A=1 TO RCDS:POSITION 5,5:PRINT A
80 GET #2,SECTH:GET #2,SECTL:GET #2,CHAR
84 REM Convert hi-lo bytes to one number.
85 SECT=SECTH*256+SECTL
90 S(A)=SECT:C(A)=CHAR:NEXT A:CLOSE #2:GOTO 1000
999 REM ** Demonstrative portion of program **
1000 REM Enter Name and address and where to store
1005 PRINT CHR$(125):PRINT "ENTER DATA:"
1015 PRINT :PRINT "(ENTER NULL LINE TO STOP)":PRINT
1020 PRINT "NAME ";:INPUT NAME$
1025 IF NAME$="" THEN GOTO 1200
1030 PRINT "ADDRESS ";:INPUT ADDR$
1035 PRINT :PRINT
1040 PRINT "WHAT RECORD (1-";RCDS;") SHOULD THIS BE"
1045 PRINT "STORED IN ";:INPUT R:GOSUB 3000
1055 POINT #1,S(R),C(R)
1060 PRINT #1;NAME$:PRINT #1;ADDR$
1070 NAME$=SPACE$:ADDR$=SPACE$:GOTO 1005
```

```
1199 REM Retrieve any particular record
1200 PRINT CHR$(125):PRINT "RETRIEVING DATA":PRINT
1205 PRINT "(ENTER 0 TO STOP)":PRINT
1210 PRINT "WHAT RECORD (1-";RCDS;") WOULD YOU"
1215 PRINT "LIKE TO SEE ";:INPUT R
1220 IF R=0 THEN GOTO 2000
1225 POINT #1,S(R),C(R)
1230 INPUT #1,NAME$
1235 INPUT #1,ADDR$
1240 ? "RECORD NO. ";R
1245 PRINT "NAME: ";NAME$
1250 PRINT "ADDRESS: ";ADDR$
1255 FOR I=1 TO 200:NEXT I
1260 GOTO 1200
2000 CLOSE #1:CLOSE #2
2010 END
3000 REM Subroutine for padding string with
3001 REM spaces so that sector pointers on
3002 REM diskette do not get scrambled.
3010 NAME$(LEN(NAME$)+1)=SPACE$
3030 ADDR$(LEN(ADDR$)+1)=SPACE$
3040 RETURN
3999 REM Establish an index file for the data file
4000   TRAP  40000:PRINT   CHR$(125);:POSITION   7,3:PRINT
"CREATING NEW INDEX FILE"
4010 CLOSE #1:CLOSE #2
4020 OPEN #2,8,0,"D:INDEX.DAT"
4030 OPEN #1,8,0,"D:FILE.DAT"
4040 FOR A=1 TO RCDS:POSITION 5,5:PRINT A
4059 REM NOTE the location about to be written to
4060 NOTE #1,SECT,CHAR
4070 S(A)=SECT:C(A)=CHAR
4079 REM PRINT the name and address to data file
4080 PRINT #1;NAME$:PRINT #1;ADDR$
4085 REM Convert SECT to hi-lo byte numbers
4085 SECTH=INT(SECT/256):SECTL=INT(SECT-SECTH*256)
4089 REM Put the pointers into the index file
4090 PUT #2,SECTH:PUT #2,SECTL:PUT #2,CHAR
4099 REM Do next record and re-run the program when done
4100 NEXT A:CLOSE #1:CLOSE #2:GOTO 30
```

```
1 REM CIRCLEZ: Compute and draw a
2 REM circle on the screen.
3 REM EZ/JP 9/81
4 REM ******************************
9 REM SELECT GRAPHICS MODE 3-8
10 PRINT "GRAPHICS MODE ...";:INPUT G
20 GRAPHICS G:COLOR 1:POKE 765,1
21 REM location 765, 'fildat', tells
22 REM the fill command which color
23 REM to use.
30 PRINT "CENTER (X,Y)...";
35 INPUT XO,YO
40 PRINT "RADIUS ...";:INPUT R
50 R2=R*R
59 REM plot 1st point on circle
60 PLOT XO,YO+R
65 REM Draw right half
70 FOR Y=R TO -R STEP -1
75 REM Compute drawto-point
80 X=SQR(R2-(Y*Y))
85 REM 'fudge factor' correct for
86 REM uneven pixel size
90 X=X*8/7
100 DRAWTO XO+X,YO+Y:NEXT Y
120 FOR Y=-R TO R:REM left half
125 REM include fudge factor:
130 X=SQR(R2-(Y*Y))*8/7
140 PLOT XO-X,YO+Y:REM plot points
149 REM top or bottom pixels?
150 IF ABS(Y)=R THEN GOTO 170
155 REM fill as you go:
160 XIO 18,#6,0,0,"S:"
170 NEXT Y
180 GOTO 10:REM try another one
```

(Continued from Page 16)

The new XL computers only have two joystick ports. The Port B is used by the operating system to manage the extra memory. Any lines that affect Port B should not be entered.

Explanation

This program sets all 16 bits for output, then puts the numbers 0 to 255 out to each port. Port A is connected to the left two jacks while Port B is connected to the right two jacks.

Line 100: This is the address of both the Direction Control (input or output) and the Data Registers for Port A.

Line 110: Direction Control and Data Registers for Port B.

Line 120: Bit two of this register determines whether Port A is the Direction Control or Data Register

Line 130: Same as line 120 but for Port B.

Line 140: ALLOUTPUT is set to all ones. SETDIRECTION has bit two reset while SETDATA has bit two set.

Line 150: Since PACTL is initialized to decimal 60 by the operating system, this line resets bit two. This causes Port A to address the Data Direction Register for Port A (instead of the Data Register).

Line 160: This sets the Port A Direction Register to all output (every bit that has a one corresponds to an output line). This will cause all eight lines in Port A to be output lines.

Line 170: This causes PORTA to address the Port A Data Register. With this configuration data POKEd into PORTA will be reflected in the voltages applied to the controller jack pins.

Line 180-200: The same as 150-170 but for Port B and PORTB.

Line 210-250: These lines demonstrate that data can be output by simply poking it into PORTA & PORTB. This sequence causes both ports to "count" from zero to 255.

Controller Jack

```
\ 1 2 3 4 5 /
\ 6 7 8 9 /
```

1. (Joystick) Forward Input
2. (Joystick) Back Input
3. (Joystick) Left Input
4. (Joystick) Right Input
5. B Potentiometer Input
6. Trigger Input
7. +5 volts 20 ma (TTL load)
8. Ground
9. A Potentiometer Input

BECOMING A MEMBER

Membership dues for the groups are $15.00 a year, which includes a subscription to Current Notes. Dues are payable at the beginning of each calendar year. Dues for new members joining during the year are reduced $1.00 for each month which has passed since the first of the year. For example, if you join in June, your dues would be $10.00. Dues may be paid at any meeting, or sent to the treasurer of the club of your choice.

DC USERS GROUP: Allen H. Lerman, 14905 Waterway Drive, Rockville, MD 20852, (301) 460-0289

NOVATARI: Curtis Sandler, 7213 Idylwood Court, Falls Church, VA 22043, (703) 734-9533

AURA: Richard Stoll, P.O. Box 7761, Silver Spring, MD 20907, (301) 946-8435

To enroll as a new club member or renew your membership, please return the card below to the treasurer of the club of your choice. The treasurer will pass your name and address on to Current Notes and you will be placed on our mailing list.

A T A R I U s e r G r o u p
Membership Form

Name _____

Address _____

_____

Phone: (_____)_____

Check the Group you wish to join:

DC_____ Novatari_____ Aura_____

I wish to receive (continue to receive)
CURRENT NOTES free of charge as a member
of the above group.

_____ _____
Signature Date