NIAD Northern Illiana ADAM User's Group
8125 W. Catherine, Chicago, IL. 60656
Phone #: (312) 775-0409

MULTI - FUNCTION USER GROUP
MONTHLY NEWSLETTER - PUBLIC DOMAIN LIBRARY
DISCOUNT BUYING SERVICE FOR HDW & SFTW

Issue #93 - December / January 1993
Editor: James Notini

Hours: Mon - Fri 9am - 9pm, Sat - Sun 9am - 5pm CST

Single Copy: $3.00
Membership Rates: U.S. 1st Class: $22.00
Canada 1st Class: $26.00
Foreign Countries: $40.00

FROM THE EDITOR'S DESK

by Jim Notini

I start this month's editorial with some news from the owner of N.I.A.D., Lyle Marschand:

> N.I.A.D. has been struggling for some time to decide what we should do about the annual membership. The current annual fee of $22 U.S. and $26 Canadian does not cover our costs for printing and mailing the newsletter especially when the amount of time it takes to write, compile and lay out the newsletter is factored in. The quality and comprehensiveness of the N.I.A.D. Newsletter has always been out top priority and our members have indicated that we are the best at providing ADAM news and articles.
>
> The first class postage rates are likely to once again rise according to what we have been reading which would force us to raise the membership rates and this is something that we do not want to do now or in the future (we have also held the line in the past).
>
> Hence, beginning with this issue, N.I.A.D. will be published every other month or six issues a year. Publishing a monthly newsletter the size of N.I.A.D.'s is a huge job. I would say that most of you have a tough time reading the entire newsletter and applying what we provide in each issue before the next issue arrives. We hope you agree with our decision to not sacrifice the quality or comprehensiveness of the N.I.A.D. Newsletter and to not raise the annual membership dues.
>
> Lyle Marschand

I am sure many of you are a little shocked by this news and I wish it didn't have to go this route, but in the long run it should benefit all our members since we will have more time between each newsletter to write articles and review products instead of always rushing to stay on schedule. Consolidation seems to be the in-thing now-a-days with large companies, but when a small company like ours is hit by hard times there are two options available, consolidate or close. The thought of shutting down N.I.A.D. was one of our options, but we still feel the ADAM Computer is a very reliable system and there are too many people that count on us to provide the supplies necessary to put their ADAM's to use. We can only hope that you understand our decisions and will continue to support N.I.A.D. and in return let us continue to support your ADAM needs.

As some of you have noticed the November '92 issue did not contain a Product List or an Order Form and unfortunately I failed to mention this in my editorial, therefore we had some letters and calls concerning this. We were trying everything at the time to find ways to continue publishing the newsletter ten times a year and even cutting out these pages did not prove to be the answer.

Aside from all that, I am sure you noticed that we are still a little behind schedule with this newsletter due to the decisions which had to be made. But beginning with the next issue, February / March, we will be back on schedule and stay on schedule. The newsletter will be mailed at the beginning of the second week of each odd numbered month and any specials contained within the newsletter will be valid until the following issue is released unless otherwise noted.

The year 1992 was a very trying time for N.I.A.D., but we have pulled through it as best we could and continue to serve the ADAM community to the best of our abilities. We feel that 1993 will serve to change things for the better and our committment to the ADAM Computer as well as community will grow even stronger as the months go by. We are always looking for new ways to better our support of this fabulous computer which so many others have given up on and would appreciate any type of feedback as to what you think needs to be improved upon, kept the same, added or even cut out. Without your feedback we will continue to run everything as we have been over the years, making small fine tuning changes here and there hoping that we are right, but never quite knowing for sure unless you let us know.

To wrap things up for this month's editorial I have a bit of good news to pass along to Illinois, Indiana, Ohio, Michigan and Wisconsin residents... NO MORE SALES TAX HAVE TO BE PAID WHEN ORDERING PRODUCTS FROM US!!

ISSUE #93 INDEX

CONTRIBUTORS: A.N.N., RICH LEFKO, JOHN TERRY, PAT HERRINGTON, DAVID SANDS, THOMAS KEENE, GUY COUSINEAU, RON MITCHELL, BART LYNCH, RON COLLINS, BOB BLAIR and BOB SLOPSEMA

N.I.A.D. SPECIALS

MEMBERS ONLY - LIMITED SUPPLIES - PHONE FIRST TO RESERVE

MICRO INNOVATIONS

FLOPPY & HARD DISK DRIVE SPECIALS

- M.I. 5 1/4" 320K ADAMnet FLOPPY DISK DRIVE $189.95
- M.I. 3 1/2" 720K ADAMnet FLOPPY DISK DRIVE $229.95
- M.I. 3 1/2" 1.44Mb ADAMnet FLOPPY DISK DRIVE $269.95
- M.I. PowerMATE 20Mb IDE HARD DISK DRIVE $269.95
- M.I. PowerMATE 40Mb IDE HARD DISK DRIVE $359.95
- M.I. PowerMATE SLOT #2 HARD DRIVE INTERFACE $24.95

NOTE: Phone Orders Only are note required on the above products from Micro Innovations. Only the products listed below and on page 3 require a phone order first to reserve the product(s).

A.H.A.P. BUNDLE

N.I.A.D. has available one ADAM HOME AUTOMATION PACKAGE available with a DIMMER SWITCH and APPLIANCE SWITCH MODULE and to top it off an EVE SP-1 SERIAL / PARALLEL INTERFACE with both SERIAL and PARALLEL CABLES. Also included is a software bundle that includes T-DOS V4.59, Easy Printer Patch (for accessing a parallel printer through the parallel interface with SmartWRITER, SmartBASIC and other Coleco software titles), A.H.A.P. Control Software and more. The EVE SP-1 will also allow you to hook-up a dot matrix printer, external modem or 80 column terminal to your ADAM system as well as the ADAM Home Automation hardware. Only one is available at the price of:

$150.00 - FIRST COME, FIRST SERVE

SEGA MASTER SYSTEM

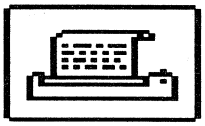
Sega 8-Bit Master System with Control Pad, Control Stick, Light Phaser, 3-d Glasses, 25 games: Afterburner, Alien Syndrome, Astro Warrior, Double Dragon, Gangster Town, Global Defense 3-D, Golvellius, Hang-On/Safari Hunt, Marksman/Trap Shooting, Miracle Warriors, Rambo II, Rambo III, Outrun, Parlour Games, Posedien Wars 3-D, Quartet, R-Type, Shanghai, Shooting Gallery, Space Warrior, Teddy Boy, Thunderblade, Vigilante, Wanted, World Grand Prix. All for only:

$150.00 - FIRST COME, FIRST SERVE

PAY POSTAGE - IT'S YOURS

2 complete ADAM systems, each include a 64K Memory Expander, 2 Digital Data Drives and ADAMLink Modem. Also: 3 ADAM Disk Drives; Zenith Amber Monochrome Monitor - no sound; EVE SP-1 Parallel Interface - all necessary software included. Set of replacement hardware for ADAM Memory Console including power supply, keyboard and 2 Digital Data Drives. ColecoVision Game System with power supply and hand controllers; 100 single-sided and 20 double-sided blank disks with desk-top storage cases. Software (all including original documentation): SmartBASIC; BNDV Vol. #1 to #7; UNDV Vol. #1; MultiWrite; Hacker's guide to ADAM Vol. I with program tape; SmartFILER V27D; ADAMCalc; Quickopy V1.1; CP/M 2.2 & Assembler; CNDV Vol. #1 to #11; dBASE II V2.43 (includes tutorials and sample programs); Secretary; FreeFiler; and Screenchop. Also, all N.I.A.D. backissues from 1985 thru 1988. Wish to get all this into the hands of someone who can use it so I am only asking for shipping costs! Call Soren Hauge in Madison, WI at (608) 231-3546.

N.I.A.D. PROCEDURES



⇨ N.I.A.D. is published bi-monthly and mailed at the beginning of the second week of each odd numbered month by the Northern Illiana ADAM User's Group. Individual issues may be purchased for the current months or a backissue for $3.00 (always check Product List for current pricing). The Dec. / Jan. issue of N.I.A.D. is the 93rd issue published by N.I.A.D., there are 92 preceding issues. When ordering backissues, please specify the number of the issue, month and year.

⇨ The standard membership rate for 6 issues is **$22.00 USA First Class** and **$26.00 Canadian First Class and it's possessions.** Contact us for membership rates outside of these areas.

⇨ N.I.A.D. welcomes contributions of original reviews, programs, articles, questions, suggestions and comments. Please include a SASE (Self-Addressed-Stamped-Envelope) if you want a written reply. Also, any contribution sent in on DDP or DISK will be eligible to receive a Public Domain program or volume in return at no charge!

⇨ Your N.I.A.D. member ID number is on the first line of your mailing label (affixed to the newsletter). The first four digits are the month and year of the final issue in your current membership. Please check this number each month to insure that issues are not missed.

⇨ N.I.A.D. will not be held liable for any issues missed due to an address change which we are not informed of. Please send this information to us as soon as possible so as not to cause any type of difficulties. Also, include your member ID number any time that you send us any kind of letter, package or order.

⇨ N.I.A.D. accepts advertising for ADAM related products and services. Cost is $35 for a half page ad and $60 for a full page ad for one month. Contact us for multi-issue discounts. Well over a thousand ADAM owners receive our newsletter each month and many more get to see it second hand. You may send in your ad in either SmartWRITER, SpeedyWRITE, PowerPAINT, PrintWORKS, other ADAM formats, IBM ASCII or IBM PUBLISH IT! DTP format files or even supply us with a high quality print out for reproduction in the newsletter. N.I.A.D. reserves the right to not advertise certain products or services which may be offered.

⇨ If **1292** or **0193** are the first four digits in your member number, this is the last issue you will receive in your current membership, It is time to renew your membership to insure that you do not miss an issue.

⇨ N.I.A.D. welcomes software developers to submit their programs for us to evaluate for possible commercial sale. Send in your products for us to test and to review in the newsletter. N.I.A.D. offers a 50 / 50 split of the sale price on all items that we handle distribution of for all developers. You will find that this is one of the best offers around in the ADAM community. We will also publish a review on the product as soon as possible and handle distribution of demo copies.

⇨ **We have exercised due care in the preparation of this newsletter. No warranty, expressed or implied with regard to the information contained herein is given, either by interpretation, use or misuse. The opinions expressed herein do not reflect those of the editor or staff unless noted.**

N.I.A.D. NEWS & UPDATES

NEW YEAR'S CLEARANCE SALE

	WAS	SALE
EDUCATIONAL SOFTWARE:		
● ELECTRONIC FLASHCARD MAKER by Coleco	$9.95	$3.95
● E.F.M. FLASH FACTS: HISTORY by Coleco	$6.95	$2.95
● STATES RACE by Hal Weber Software	$15.95	$5.95
ENTERTAINMENT SOFTWARE:		
● ADDICTUS by Reedy Software	$19.95	$9.95
● DRAGON'S LAIR by Coleco Electronics	$16.95	$6.95
● PHRASE CRAZE by Reedy Software	$19.95	$9.95
● PHRASE PAK i by Reedy Software	$12.95	$4.95
● PHRASE PAK ii by Reedy Software	$12.95	$4.95
● REEDY ENTERTAINMENT PACK by Reedy Software	$15.95	$5.95
● STAGE FRIGHT by Reedy Software	$15.95	$5.95
● STRATOZAP by Allied Creative Engineers	$16.95	$6.95
● SUPER ZAXXON by Coleco Electronics	$9.95	$4.95
● TEMPLE OF THE SNOW DRAGON by Digital Adv.	$19.95	$9.95
● U-MATCH-EM by Phoenix 2000	$14.95	$4.95
● VASE OF TURR, THE by Walters Software Co.	$19.95	$9.95
GRAPHICS SOFTWARE:		
● BOLD GLORY by Eyezod Graphics	$16.95	$8.95
● COLECO GRAPHICS PROCESSOR CART by Coleco	$39.95	$19.95
● NORMAN'S RAILROAD by Norman's Software	$14.95	$4.95
HOME & BUSINESS SOFTWARE:		
● APPOINTMENT BOOK by The Maine ADAM Library	$24.95	$9.95
● INVOICER III by ADAM's House	$19.95	$9.95
● LABEL WORKS, THE by Walters Software Co.	$24.95	$14.95
● MisSPELLER by Walters Software Co.	$9.95	$4.95
● PRINTWORKS, THE by Walters Software Co.	$24.95	$14.95
● RECIPE FILER by Coleco Electronics	$9.95	$4.95
● SHOWOFF II by Digital Express Inc.	$14.95	$4.95
● SmartBASIC V1.x by Drushel Software	$29.95	$14.95
● SmartFILER by Coleco Electronics	$9.95	$4.95
● SmartLETTERS & FORMS by Coleco Electronics	$9.95	$4.95
● SmartLOGO by Coleco Electronics	$15.95	$5.95
● SmartTERM V1.02 by Keheo Software	$15.95	$5.95
● TAX HELPER 1991 by Hoosier Software	$19.95	$4.95
● VIDEOTUNES by FutureVision	$24.95	$14.95
MEDIA UTILITY SOFTWARE:		
● ADAM'S TOOLKIT by Walters Software Co.	$24.95	$14.95
● BACKUP 3.0 by M.M.S.G.	$10.95	$4.95
● COPYCART+ V2.0 by M.M.S.G.	$19.95	$9.95
● DECIMAL DISASSEMBLER by Walters Software Co.	$19.95	$9.95
● E.O.S. PROGRAMMING KIT by Walters Software Co.	$29.95	$14.95
● RAMBOOT by Walters Software Co.	$19.95	$4.95
ADAM BOOKS:		
● ADAM PROGRAMMING GUIDE by The M.A.L.	$15.95	$9.95
● ADAM GLOSSARY by The A.N.N.	$4.95	$2.95
● ADAM SURVIVAL GUIDE by The A.N.N.	$24.95	$14.95
● BASIC PROGRAMMING TUTORIAL by ADAM's House	$14.95	$9.95
● BEST OF ELECTRONIC ARTS MANUAL	$4.00	$2.00
● FROM BASICS TO BASIC WITH ADAM by Road.	$19.95	$9.95
● HACKER'S GUIDE TO ADAM VOL. I by Hinkle Public.	$11.95	$5.95
● HACKER'S GUIDE TO ADAM VOL. II by Hinkle Public.	$11.95	$5.95
● LEARNING TO READ WITH ADAM by Road. Publ.	$24.95	$14.95
● UNCOMM. DIS. OF ADAM E0S7/SB V2.0 by Road.	$24.95	$4.95
● UNCOMM. DIS. OF SmartBASIC V1.0 by Road. Publ.	$15.95	$4.95
ADAM SUPPLIES:		
● ADAM 256K DIGITAL DATA PACK	$2.00	$1.75
● ADAM 256K RIGHT DIRECTORY DIGITAL DATA PACK	$1.00	$ .50
● ADAM DUST COVER: DISK DRIVE	$8.95	$4.95
● ADAM DUST COVER: EXP. MOD. #3 CONSOLE	$8.95	$4.95
● ADAM DUST COVER: MEMORY CONSOLE	$8.95	$4.95
● ADAM DUST COVER: SYSTEM PACK	$21.95	$11.95
● ADAM PRINT WHEEL: COURIER	$5.95	$3.95
● ADAM PRINT WHEEL: PICA	$5.95	$3.95
● ADAM PRINTER RIBBON	$5.95	$4.95
● PANASONIC KX-P115 or KX-P145 RIBBON	$12.95	$6.95
ADAM HARDWARE:		
● ADAM KEYBOARD WITH 7ft. ADAMnet CABLE	$24.95	$9.95
● ADAM PRINTER POWER SUPPLY	$49.95	$24.95
● ADAM TAN HAND CONTROLLER	$4.95	$2.95
● ADAMLink 300 BAUD MODEM	$49.95	$24.95
● ADAMnet 7ft. CABLE	$2.95	$1.95
● EXPANSION MODULE #2 WITH TURBO CART	$44.95	$24.95
● M.I. 64K MEMORY EXPANDER	$29.95	$22.95
● M.I. PARALLEL PRINTER CABLE	$14.95	$10.95
● M.I. RS-232 SERIAL CABLE	$14.95	$10.95
● ROLLER CONTROLLER WITH SLITHER CART	$44.95	$24.95

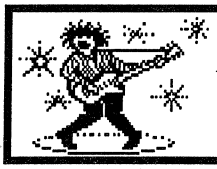
⇨ **N.I.A.D.** has assembled a tremendous array of specials on software, hardware, books and supplies for the ADAM Computer that are listed on this page. Please notice that these sale prices are valid while supplies last. We do recommend calling first to insure that products are still available, if you do not and the product is still available, your order will be filled, but if we have sold out on stock for the said product you will be issued a refund or credit.

⇨ **HELPER'S FRIEND** has been released by Bruce Walters of Hexace Software. This is the long awaited Public Domain software release that allows owners of SmartWRITER's HELPER to modify and add to the CONTROL sequence printer codes. Helper's Friend is written in fast machine code and uses windows to display menus. Also included is a thorough documentation file that should be carefully read.

⇨ **M.W. RUTH CO.**, operated by Jay and Ruth Forman, is still in the ADAM mail-order business unlike last month's report that they may have closed their doors. As we stated in the November '92 issue, we had seen a report of M.W. Ruth Co.'s closure and would check into it ourselves. It took a while to get through to Ruth Forman, but we finally did and she confirmed that they were still in business although a little hard to reach at times. Ruth did recommend mailing in orders to them if you can not reach them over the phone.

⇨ **DRAGON MASTER**, the adventure role-playing game we have been mentioning on and off since ADAMCON 04, by Scott Gordon of Magic Vision will not be released until later this summer at ADAMCON 05. This is not a bad sign at all since Scott has been busy at work adding new features and new capabilities to the original game design that should knock some socks off. As the saying goes, "Good things happen to those who wait".

⇨ **ADAM MAP: STATES VOL. #6** and **#7** have been released by Carl Harrison of Harrison Productivity. The sixth and seventh state volumes are a collection of five states each: Vol. #6 - Georgia, Florida, South Carolina, North Carolina and Virginia; Vol. #7 - West Virginia, Ohio, Maryland, Delaware and Kentucky. Watch for further state volumes coming soon. Retail price for each volume is $6.95.



THAT'S RICH!
What Happened to 1992?
by Richard Lefko

You may have been wondering (I hope) what ever happened to Rick Lefko??

Well, fact is, I'm still around, still have my ADAM, just haven't been using it very much. Now that isn't because ADAM hasn't been filling my computer needs or something like that, it's because I just haven't had the time!

By the time my son Greg will have celebrated his first birthday, and I will have reflected upon a year of wonder, gone by, like lightning.

It's been a wonderful year of discovery for both my son and me. I'm so very proud of both him and his mom, and the best part is, it's only the beginning.

Now-a-days, whenever I turn the ADAM on I begin to think about the time Greg will be sitting on my lap and we'll be enjoying all that ADAM has to offer, together. The ADAM will be Greg's first computer! Not a bad first computer if I do say so myself! Do you remember what your first computer was like??

In anticipation of that day, I bought and saved every piece of "kid" type software title that COLECO had put out. Soon I'll finally be able to start breaking out software like, Richard Scarry's Best Electronic Wordbook Ever, Dr. Seuss' Fix-Up the Mixed-Up Puzzler, the Cabbage Patch Kids games, Smurfs, etc.. Then as time passes I'll get to re-discover all those video games I haven't played in years, and that is one of the best discoveries I've made thus far. To look at the world through the eyes of a child is to re-discover it all again. Whether it's watching the wind shake the leaves on a tree branch or throwing pebbles into the pond behind my house, children make us take the time to "see" it again.

I know how difficult and tough times have been, and we often lose sight of what's really important in life as we try to survive day to day and each of us gets caught up in his/her own world.

If there was only one holiday wish I could offer, it would be this:

That you would take a moment and try to remember back to your happiest holiday memory, and share it with someone you love. If you have a child, take a look at the world once more through those innocent eyes, and remember.

WHAT THE ADAM WORLD REALLY NEEDS IS:

I've said this before, so this time I'm hoping that all 3 1/2" 720K and 1.44Mb disk drive users will band with me, by writing me, and I will make the need known to those who can help us!!

The need is this: In most Directory management software, such as "File Manager", you can create up to 8 directories. Since each directory can hold 35 files, that means you can store up to 280 files on a disk. The problem is on a 720K disk, 280 1K files (like Clip-Art files) leaves more than half the disk useless. Sure, there are programs that allow more directories, but most other programs won't print out or recognize more than 8.

I'm sure that there is some brilliant programmer out there who can help us spend some money by writing up some kind of archival type utility or perhaps allowing partitioning of some kind.

Rick Lefko
20 Ashley Dr.
Milford, NH 03055

TDOS HARD DRIVE PRECAUTIONS
Part II: Your Backup System Disk
by Guy Cousineau

EDITOR'S NOTE: Part I appeared in the October 1992 issue.

Once you have installed a hard drive on the ADAM, you are ready for speed. Particularly under TDOS, you will be amazed at how much faster programs load in and execute, especially if they work with large data files. Don't let your system grind down to a halt by disk failure! Follow these recommended procedures to minimize the impact of various crashes.

BACKUP SYSTEM DISK

If your hard drive is dead, it will do you little good to try and reboot TDOS from disk since it will try and read from the hard drive. What you need is a DISK BASED TDOS disk. If you have one of these, guard it carefully. If you don't have one, use a disk editor such as SUPERZAP to modify the 258th byte of 40TDOSxx.COM. This is byte 2 of the sector 2 which is normally a ZERO. Changing it to a non-zero value will tell TDOS to ignore the HARD DRIVE as a device even if it is found. You can then run the modified 40TDOSxx.COM to install the TDOS system on disk. Don't forget to re-change the byte to ZERO after this process. Once in a while, you should boot the resulting disk based TDOS disk to make sure it works correctly.

Now that you have a backup system disk you need to put a few files on it. Following is the minimum requirement along with reasons:

- **80TDOSxx.COM** - only if you have an 80 column card
- **40TDOSxx.COM** - in case your 80 column goes on you
- **HDINST.COM** - to reformat the drive
- **HDDIAG.COM** - to test or restore the drive
- **SPZ.COM** - or another disk editor in case you need to poke around
- **MAINT.COM** - or whatever you need to get at your backups
- **SDCOPY.COM** - if you have only one disk drive

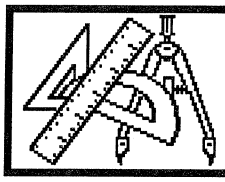
The other essential item you will require is the CONFIGURATION information for your hard drive. This includes: Number of Heads, Number of Tracks, Seek Type, Write Precompensation, MEGS Reserved for E.O.S., Number of E.O.S. Drives

You should have written down this information when you initially configured your system. If you did not, DO IT NOW! Print it out and insert the paper in the sleeve of your backup system disk.

CRASH RECOVERY

If your hard drive stops working, just pull out your backup system disk and reboot TDOS. Then run the HDINST.COM and answer the questions just the same as before; this will ensure that your system is reconfigured exactly the same way. When you get to FORMAT YES or NO, answer 'n' for NO and proceed with the partition information. At the conclusion of the program, a new block zero will be written to the hard drive. Try rebooting it again and cross your fingers.

If the first procedure does not work out for you, use SPZ or another disk editor to look at TRACK 0 of the hard drive. If you can read and write to it then your problem lies elsewhere. Try BOOTING the E.O.S. system and the TDOS system to isolate the problem area. More than likely, the only way to solve these problems will be to reformat the entire drive. Repeat the procedure above and this time answer YES to the FORMAT question. You will then have to REINSTALL TDOS and re-run the E.O.S. HARD DISK INSTALL to install the BOOT and INITIALIZE your logical drives.



FUNDAMENTALS OF COMPUTER PROGRAMMING

Chapter II, III & IV: Housekeeping, Output & Input

by John Y. Terry, Sr.

CHAPTER II: HOUSEKEEPING

As the term "housekeeping" implies that the house is to be put into order, the computer needs to be put into order by the program. This is the section of the program in which any program identification should occur. Holding areas and counters are established. Page headers and date routines are here, as well as page counters. This is the area in which all of the "housework" of the program is done.

This section is referenced from all of the program sections, and is normally written in conjunction with other program sections.

All programs should have every instruction numbered. Some computers support un-numbered instruction sets, and numbering instruction is left to the programmer's discretion. Be discreet. Use instruction numbers! If instruction numbers are not required by the computer, these numbers should be used for the programmer. These instruction numbers are the programmer's reference points throughout the program.

Program instruction numbers must be sequential. When assigning program instruction numbers, it is usually best to number using a system that allows the insertion of other instructions at a later date. Numbering in increments of ten seems to work very well. When numbering by groups of 10, the second will be 20, and so forth. This will allow for the insertion of up to nine additional instructions at a later date if it becomes necessary. Allowing for additional instructions from the beginning will avoid the cumbersome and time-consuming task of renumbering and entire program if additional instructions need to be added later.

The various program sections should also be considered for special number groups.

Reserve the first 1000 instruction numbers for housekeeping, the next 2500 for Input, etc. In this way, each program section is easily identified and the individual instruction number will immediately point to the proper section.

REMARKS should be freely used. These are notes to yourself or to some other programmer, telling what you are attempting to do in that particular section of the program or that instruction.

When having to make a change in a program or when changing some other programmer's work, help is not only appreciated, but needed. Keep remarks short and concise. Although remarks do not compile along with program instructions, they DO take up space.

It is suggested that the five program sections be individually identified with a remark statement. If the section identification is enclosed with asterisks or some other symbol, the section names of titles will stand out and be easily recognizable. EXAMPLE:

```
10 REM *** IDENTIFICATION SECTION ***
20 REM Program Name
30 REM Programmer's Name
40 REM Date Written
2500 REM *** INPUT SECTION ***
```

Headings and titles should be laid out in this section of the program. Use a blank sheet of paper (quadrille paper with 1/10 inch squares is best, as that is the standard print position size) and "design" the headings and titles that will be needed, and then code them accordingly.

Use X's to fill in all alphabetical spaces within a data field, and 9's to fill in all of the spaces within a numerical data field. Use a combination of X's and 9's to fill in all the spaces within an alphanumeric data field.

CHAPTER III: OUTPUT

The program's output must be determined before input can be established. On paper that was used to lay out the heading and title information, write in what output is wanted from the computer, again using X's and 9's as before.

All of the output must be established, and its position on the page or screen determined, and programed. If any output is to be to disk or data pack, this must also be programed in the desired format.

Remember, when determining the output, that not all of the input data need be output. Use only what is needed.

When programming the output element, do not be concerned with the input or any other section of the program except the housekeeping section. Always keep the housekeeping section in mind.

Program the output section from the previously prepared draft of that section. Always work from a written draft. It is easier to do correct work from a written medium than it is from the "top-of-the-head".

Work slowly and carefully. Be sure that you are writing the instructions in their proper format and that they are in the proper sequence for your programming language.

CHAPTER IV: INPUT

Once the computer output has been established, the question arises as to where this output is going to come from.

The answer of course is from the input and data manipulation sections of the program.

When determining the input, it must be established if the input is to be from disk, tape, keyboard, modem, etc. Each of the input devices must be considered separately and be programed according to the input devices' specific requirements.

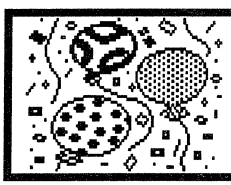
The inputed data must either be used immediately or stored for use at another time.

The holding areas and counters established in the housekeeping section of the program are used for this purpose.

It is best to have all input data available on the system prior to doing data manipulation. Sometimes the result of one data manipulation is the input for another data manipulation. This possibility must be considered.

At this point in the program, other sections of the program may be used. Some inputed data may need to be immediately output to disk, print, etc. Keep the use of other program sections as limited as possible to maintain the integrity of the program and to facilitate debugging or changing the program later.

Care must be taken to insure the logical flow of the program from start to finish. Remember, the computer will do exactly what it is told to do when it is told to do it. If the program is not in a logical form, the results will be unpredictable and difficult to fix later.



MOVE & COPY TUTORIAL
for YULE TOOLS and PowerPAINT
by Pat J. Herrington

(A.N.N. Notice) The following article is a tutorial for using the MOVE/COPY function of PowerPAINT. It first appeared in the documentation for Yule Tools I by Eyezod Graphics. It may be reprinted in user's group newsletters per Eyezod Graphics. Be aware that it contains some references that are specific to Yule Tools I. The documentation originally contained that are not contained in this article.

If you are not yet familiar with all the functions of the MOVE/COPY key, now is the time to learn! You will get infinitely more use from your YULE TOOLS and PowerPAINT when you know how to manipulate all the design elements.

To access this menu, simply press the MOVE/COPY key. You will then be presented with several options.

"COPY CELL" is used to place the entire screen of the current cell in one or more different cells. This is very useful for keeping interim versions of work in progress, for making experimental changes, and for creating headers and footers. Select the number of the cell you wish to copy to by pressing the corresponding number on the keyboard (1 to 4, or 1 to 8, depending on the size of your memory expander.) You can copy to as many cells as you like. Each time, you will be shown the contents of the new cell, and asked whether you really wish to replace it. When you are done, press ESCAPE to return to the MOVE/COPY menu.

Another useful option is MOVING WINDOW. This helps you to see how the contents of your workspace line up in relation to each other. You can scroll between cells, moving up, down, left, or right, depending on your current cell. Use the arrow keys to scroll, and the ESCAPE key when you are done. No changes are possible from this function.

PULL PICTURES allows you to move the entire screen in any direction, or to move just one 8-pixel row at a time. You probably will not use PULL BY ROW very much, but it is handy for centering text.

PULL (entire) SCREEN is very convenient for erasing large areas of graphics around any of the four edges of your screen. For example, if you wish to eliminate the text on the bottom of the CANDLES: picture, use the DOWN arrow key to pull the picture to the bottom of the screen until the text disappears. Then use the UP arrow key to pull the candles back in place. MOVE SLOWLY. The screen moves 8 pixels at a time, and the results do NOT appear on the screen instantly. Holding the arrow key down for too long can easily erase the entire picture. If this happens, don't panic. TO GET YOUR PICTURE BACK, PRESS THE <UNDO> KEY!!!

PULL SCREEN also helps you to center graphics between cells. If you would like to place a picture in the middle of a letterhead, and then add details to either side of the picture, you first copy the cell to the adjacent cell. Then use PULL SCREEN to pull the picture in Cell 1 to the right, and, after returning to the Primary Menu and switching cells, use the same technique to move the contents of the right cell to the left. Your screen is now centered, and you can add any details you like. The GARLAND: pictures were designed this way. If you press the left or right arrow key 15 times, you will move your picture exactly half way across the screen. You can use the same technique to center vertically with the up and down keys. (To move half the screen VERTICALLY, press the key only ten times.)

The other option in this menu is SET FRAME. When you press this key, a small shape appears which represents the location of the upper-left hand corner of your frame. You can move it across the screen with your arrow keys, and press <RETURN> to fix it in place. Then another shape appears, representing the lower right-hand corner. Use the arrow keys and <RETURN> to finish your frame. Whatever is inside the frame can now be moved, copied, or erased. You are limited to 8-pixel increments. Your frame can be square or oblong, but can be no smaller than 8 pixels square and no larger than 64 pixels square.

You are now presented with three options. ERASE will eliminate the contents of the frame: all the foreground details will be gone, and the background will be changed to default black.

If you do not want black background, you can change it later from any of the three menus: GLOBAL COLOR, DRAW BACKGROUND, or even INSERT, using the space bar.

Rather than ERASE, you can choose to MOVE or COPY the contents of the frame. MOVE will put them in a different area, erasing them from the original location. COPY will leave the original graphics but place another copy of them in a new location. In either case, you choose the new location (in 8-pixel increments) with the arrow keys and fix in place with <RETURN>. You can MOVE or COPY to a different cell, too. Hold down the Control key while pressing either U or D (for Up or Down.) Control-U will move to a higher-numbered cell, and Control-D will move to a lower-numbered cell. When your frame is in the correct cell, use the arrow keys to position it and <RETURN> to fix it permanently. You will be asked whether or not the results are okay. If you press Y, the change is permanent. Then you will return to the original cell. If you press N, you will return to the original cell, but nothing will have happened. You can start again if you like.

Think of the contents of your frame as being Clip-Art that you choose yourself from portions of your screen. The difference is that the "clip" can be of different sizes. (The minimum size is the same as a font letter, and the maximum is as large as a regular clip.) This flexibility means that you can choose just a small portion of a clip and place it in various locations in your workspace (in every cell, if you like.) For example, you can set a frame around each individual ornament in the "frame clips" on this volume (one at a time) and move it to a new location.

One word of caution: do NOT use the MOVE function unless you are moving to an entirely new location. If any portion of the new location overlaps any portion of the old location, the contents of the frame will be ERASEd instead. When in doubt, use the COPY function. Then you can follow up with ERASE if you still need to get rid of anything. Or you can erase by using the space bar in the INSERT menu.

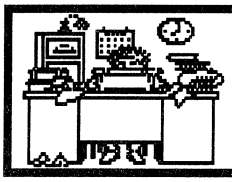
Remember that the correct keys for moving or copying the contents of a frame to a different cell are Control plus U or D, NOT Control plus the arrow keys. Don't be intimidated by references to "8 pixel increments". You don't have to count pixels.

The frame will automatically count off the increments. Play with it and find out what you can and cannot do.

If you ever become frustrated because something you've drawn does not fall within the increments, you might want to know that there is a way around it. If you SAVE your picture as a SmartPAINT file, you can use a Public Domain program by Digital Express called SimplePAINTER to replot the picture a few pixels. Though SimplePAINTER is not as user-friendly as most DEI programs, it is free, and it works.

There are many reasons you might want to move portions of your screen to another screen (besides the obvious reason that you might just WANT them there.) For example, you might want to change foreground or background colors on only one portion of your picture.

(CONTINUED ON PAGE 14, COLUMN 1)



BEYOND SmartWRITER

Part IV: The VDE Story

by David Sands

VDE is like the fellow, who is the featured speaker at a convention: somebody always stands up to say he needs no introduction, and then proceeds to introduce him.

You've heard about VDE somewhere; it's Video Display Editor version 2.66. It's the final CP/M - TDOS version of a small, fast word processor, that's been constantly improved since it first began life, in 1984, the same year as ADAM. Better known as VDE266 or more simply as VDE, it has a few talents that haven't been previously discovered in this series. And, it is becoming a big item in the ADAM world, mainly because it is fast. The ADAM isn't usually thought of as a fast machine, but VDE will make you think it is.

Speed is definitely one of VDE's virtues -- in the ADAM world, VDE is the greyhound, lean and built for speed, SpeedyWRITE is a collie, gorgeous and elegant, quick enough but not really a racer, and SmartWRITER, well, SmartWRITER's an old dog. This overview of VDE will be broad and shallow: broad enough to sketch out the many attractions of this particular program, and yet shallow as well, because both the ADVISA and ANN amongst others are carrying Tom Keene's detailed tutorial series on VDE and I am not about to try to duplicate it, and because that's the kind of guy I am.

If you never use VDE it will probably be because of the T word. As in T-DOS. Right, VDE is a T-DOS word processor, and that means Trouble right here in River City, folks, and that's trouble with a capital T, capital D, capital O, capital S and that spells to "hellwithit" so far as many ADAM owners are concerned.

Which is too bad, because T-DOS is just an operating system and its only influence on VDE266 is that you first load it, type "VDE" and then work in VDE.

If ten million secretaries can go to work every work day morning, turn on their MS-DOS computers, get Word Perfect on the screen and start typing, you and I can run T-DOS and VDE. The procedure is almost identical, and all they have to look forward to are coffee breaks whereas you and I have the excitement of learning a marvellously fast and versatile word processor. And besides, TDOS gives you access to all the CP/M programs in the world, many of which have no E.O.S. counterparts.

Trust me, it will be worth it. VDE, as I have said, is far faster than anything you can imagine on the ADAM.

What are the downsides; you've give us the good news, now hit us with the bad; level with the people; don't kid the troops; -- I can hear you all muttering out there.

It's an election year in the USA, and Canada's various politicians have written us and themselves a brand new constitution, and you want honest reporting.

Okay. You know that T-DOS (CP/M) VDE is a control code word processor; it has about 74 of them, and a set of options as well. It does not use SmartKEYs, and it has few, and brief, error messages. Depending on the version you have, it may not use the arrow keys (but the ADAM version commonly available does), it doesn't automatically reformat its paragraphs if you insert or delete, and its display is strictly business.

There is no page length marker at the side of your TV screen, no SmartKEY display, no bells, or whistles or little burps like SmartWRITER. Also, when you use VDE with your ADAM printer, unless you "install" them with VINST266.COM, some of the features you may have become accustomed to, such as your backspace key, delete key and underline key are not there. However, if you do use the original ADAM printer, with its limited repertoire VDE offers you the plain vanilla printing, that you've had from the beginning with ADAM, but with much more formatting flexibility and the faster editing speed we have already mentioned. A good combination.

VDE doesn't offer print justification, but a separate program, VDEJUS, by Guy Cousineau, is available to do it. VDE was originally a CP/M word processor able to be used on any of the many different computers using that system. They all had one thing in common, though, that ADAM somehow missed out on. And that was an 80 column screen. Which means that you work in the equivalent of Moving Window format, and your screen will zip to the right on every Control key press. You use Control-left arrow to zip back. I like Moving Window, and I set up VDE with its horizontal margins set so a text fills my screen without sideways scroll, just like SpeedyWRITE.

Formatting and reformatting for printing is easy to do, and if I haven't mentioned it previously, quick.

Of all of VDE's drawbacks, the control codes are the most relevant. They have to be learned, but it helps that many are mnemonic -- their letters relate to the name of the function. ESC-F gets you the directory of Files, Control-OR sets the Right margin, ESC-S Saves.

Some others you just have to remember... For example, Control-C jumps you one screen ahead in your file.

VDE comes with a large DOC file, VDE266.DOC, which will print out a 21 page manual, as well as a "Cheat Sheet" document file, called VDE266.QRF, which gives almost everything you need on one page.

There are some close similarities between SpeedyWRITE and VDE in terms of operations and features, and VDE is often described as a look-alike for WordStar. However, all word processor programs have so many similarities that someone like me, who's learned four or five this year, starts to have deja vu all over again every time he hits the keyboard.

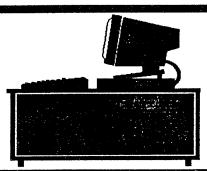
VDE was and is used for writing programs: it has "non-document" modes available for that, and ASCII and "W" type WordStar text file formats. VDE's documentation is complete, easy to read and follow.

If you've grown old scrolling through your SmartWRITER documents, if pressing SmartKEY VI to confirm that you do, really - really do, want to Delete, has lost its charm, and if SpeedyWRITE's small file space is cramping your style on your PhD thesis, then VDE has a lot to offer you. In VDE, Control-QR has you at the top of the file right now; Control-QC gets you to the bottom, fast too. Hitting Control-O+up arrow makes the line holding the cursor the top of the screen, Control-OS double spaces your file when you print it.

There are nine separate Control codes for deleting by character, word, by line, backward and forward, and undeleting any of your previously deleted characters and lines, and you do not Hi-Lite anything to Delete: you want it gone, it's history. Similarly, find and replace is swift, quick and fast, fast, fast.

All VDE users rave about its speed, and I'm surely no exception. Speed in word processing is a combination of a program's design, its size, and the convenience of its control system. In VDE you have a bare bones "necessities" design, a small (16K) machine language program, and to enable you to load all of your 40+ K file into memory, it is compressed,

(CONTINUED ON PAGE 14, COLUMN 1)



PUT IT IN WRITING!

Part III: The Grave Accent

by Thomas J. Keene

This is the grave accent (`). I use it as a surrogate to represent characters that don't correspond to those not found on ADAM's keyboard. And there are a host of these characters. For example, modern printers can print the symbols for (the British Pound Sterling, the cent sign, the section or numbered clause sign, the one-quarter symbol, Greek-based engineering signs, and French/Spanish accented vowels or German umlauts!) But you'll not find any of those on your ADAM keyboard (or on very many other computer keyboards either). After I have stored my text, and now want to replace these subterfuge grave accent characters with the printer code that produces the desired results, I use a disassembler such as DU-V87, or JKL Utilities to effect the change. I prefer DU-V87 because it makes it so easy to locate the exact characters to be replaced, including one which must be replaced with a null.

Since we now have all of these ultra-capable printers available for our ADAM, it would be unfortunate not to avail ourselves of their incredible versatility. If our ADAM doesn't have the keyboard for these exotic characters and if it can't issue a null, then we must improvise to put these characters into our text.

As I have said, I simply issue a grave accent (`) for every one of these characters which I can't enter with my keyboard. If it happens that I have a great many of these in a given text, then I must take care to note which grace accent is to be replaced with which special character.

Actually that is quite easy to do as I will explain. But first I use VDE266 to prepare the text with a grave accent substituting for a special character. After closing out the text, I prefer to make a back-up copy of the text, which is identical to the text as originally created. The reason for this is that I will use the backup copy to introduce the needed changes and leave the original text exactly as it was first written. To do this, I use VDE to re-open the text as though I were going to continue adding text. Suppose I have a file named APRIL.TXT and it is stored on my disk (or data pack). Assume that VDE266 is on my A: Drive and APRIL.TXT is on the B: Drive. This is the command I will write at the A> prompt:

A>VDE266 B:APRIL.TXT <CR>

VDE goes to work and brings the file APRIL.TXT into memory and awaits my editing. In this case I don't want to edit or change a thing, so I issue a save command (Control-K X), this saves the file to the disk unchanged. And is also returns me to CP/M. Now I have two identical files on the B: Drive disk. One is APRIL.TXT and the other is APRIL.BAK. The newest of these two files is APRIL.TXT. The previous APRIL.TXT resides on the disk in the same location that it did before this duplication. The only thing that has been changed is the filetype from TXT to BAK. It is this APRIL.BAK that I will "edit" with DU-V87.COM I will briefly go through the DU-V87 operation to cover this particular application. If DU-V87.COM is on the A: Drive, I enter this command:

A:DU-V87.COM <CR>

Presently I will see this on my screen:

DISK UTILITY V8.7
Universal Version under CP/M 2.2

Type ? for help
Type X to exit

:

The colon is the prompt in DU-V87, so at the prompt I enter the command to examine the B: Drive. This command is LB followed by a carriage return. LB may be upper or lower case. Now DU-V87 is controlling the B: Drive. If I want to have a look at the directory of the B: Drive, I command it to go to GROUP zero with the command:

:GO <CR>

If I want to see the directory in ASCII characters, I type the letter "a" followed by a carriage return. Both the G command (G stands for Group) and the display ASCII text command can be combined in a one-line command by placing a delimiter (a semicolon (;) between the two commands like this:

:GO;A <CR>

The display on the screen will be that of one sector. To have a look at the next sector I just hit the plus sign (+). I can go back a sector by hitting the minus sign (-). It isn't necessary to explore the whole directory to find where the file APRIL.BAK is stored. DU-V87 has a find command which is quite simply the letter "F".

So at the prompt, I enter this command:

:fAPRIL.TXT <CR>

Here again the command may be uppercase or lower case. DU-V87 will search the directory and will return the two line extent which tells us where it is. A typical example might look like this:

:fAPRIL.TXT <CR>

```
40 00415052 494C2020 20545854 00000058 *.APRIL TXT...X*
50 0D030F10 11121314 15181900 00000000  *..............*
```

I won't go into a detailed explanation of the CP/M extent, but I will point out a couple of salient facts. This is a look at one entry of the directory. The line starting with 50 tells us where the file APRIL.TXT is stored. The first group (1 kilobyte or 8 sectors) is stored at group 0D (in Hex). Note that the remaining groups are restored contiguously until Group 15. After that the next group is Group 18. 16 and 17 are used to store something else, which is perfectly normal for CP/M. For an easy search or the grave accent, it would be nicer if the group was contiguous. It isn't absolutely necessary but it sure is more convenient if the file APRIL.TXT was not spread all over!

The next thing I want to do is find the locations of all of the grave accents that need to be changed. DU-V87 has another marvellous searching tool that will find either a character, a word or even an ASCII code. For instance, the ASCII code for a grave accent is 60 (Hex). We can search for the first grave accent but first I must go to Group 0D where APRIL.TXT is stored. To do this I type this command:

:GoD;a <CR>

This will move to Group 0D and display the text of the contents of the first sector of that group. The search command is the equal sign. If I am searching for an ASCII character I just enter the character like this:

:=` <CR>

This command tells the program to find the first occurrence of the grave

accent character. The ASCII code for the grave accent is 60 Hex. Knowing this I could have searched for the ASCII code instead of the character. The only difference, in this procedure, is that I put the code byte in "less than" and "greater than" brackets like this:

:=<60> <CR>

Either way is okay but it is easier to search for the character than the code. When it finds the next occurrence of the grave accent, it stops and tells me that it is at some place. Here is an example of a search for the first grave accent:

:G0D;a <CR>

G=0D:00, T=26, S=1, PS=0

```
00  *.@.W1.w1.Q..E.G.*
10  *k..a..CB.N...WOR*
20  *T BEHANDLUNG...q*
30  *'..Und So Weiter*
40  *...g.WO.wO.1..Qq*
50  *.a..F.H.. *
60  * Continuin*
70  *g last month's d*
```

I am now at the start of the file and at the prompt I issue the =' command, this is what that looks like along with the response:

:=' <CR>

= AT 31

G=0D:)), T=26, S=1, PS=0

The address where the character ' was found is given in the response. This might very well be down further in the body of the text. But in this case, the sector where it was found is the current location of our program. I can see the ASCII contents of this sector by typing an "a", but in this particular example, I already have a look at this sector. The AT location is always the first character following the character or string. In this case it says that it is at 31. So look for byte number 30 (one less than the address given in the = AT response). And if I look at the ASCII printout of Sector S1 of Track 26 above, I will see the grave accent.

This needs to be changed to a null so the command for that is:

:CH30,00

Here I used one of the change commands. CH followed by an address (30 in this case) will cause a change in the ASCII code. I am going to replace 60 with 00. The other change command is CA followed by an address. In this case I will use the actual character and not the ASCII code for that character. For this application, I will likely never use that particular character.

And the reason for that is, I am changing a character which I don't have on my ADAM keyboard. There is no null on the ADAM keyboard. Remember that zero is 30 in the ASCII code. I don't want zero -- I want a null. And that code is 00 Hex. So I will almost never be using the ASCII change command (CA). At least not in this kind of an operation.

In the change command, there is a delimiter (a comma) immediately following the change address. This in turn is immediately followed by the character that is replacing byte 60 (code for ') and just as soon as the command is issued, I will see the byte that was eliminated displayed on the screen. This is, of course, the byte 60 Hex. I can check this by typing either a letter A or a letter D (followed by a carriage return). A is an ASCII dump and D is a digital dump.

For this check, I prefer the digital dump where I can see that the byte at address 30 has been changed from 60 to 00. Once I am satisfied that I have made a proper change, I will make the change permanent by typing the write command which is simply the letter W. THIS IS IMPORTANT, for, if I leave this sector without writing the change to the disk, it will remain just as it was. This is handy at times, in case I made an error in the change command. If so, just advance one sector, and then return to the original sector. This was I am back where I started and can try again. But once the write command has been issued, the change is entered on the disk, and the only way I can undo a mistake here is to change the byte with the CH command.

This may sound like an arduous procedure, but it really isn't. It's just that it takes a few words to explain it. Doing it is extremely easy. Now that the first grave accent has been changed I can search for the next one with the =' command. In those situations where I want to print an exotic character for the Extended Graphics Table (things like the British pound symbol, or the cents symbol, or any of those characters I showed earlier) I follow the very safe procedure. I find the grave accent as before and, instead of a null, I change it to the appropriate code. The British pound code is 9C; the cents symbol is 9B. If you check your ADAM CP/M manual you will find no reference to 9B or 9C. However, these are now pretty well standardized in the computer world and the codes are readily available. You are sure to find a set in the manual that came with your dot matrix printer.

Once you get the desired changes in the backup copy of your text, this file will retain it's integrity in just about every operation except editing it with VDE. You can print it, copy it to another disk, examine it with DU-V87.COM or JKL Utilities. But if you call up the file with VDE, all of the nulls you have entered will be evaporated. I caution you so you will not be shocked if you edit a file with nulls in it, with VDE, those all important nulls may not be there to perform some of the embedded printer commands. If you have changed your user options so that a backup copy is not automatically made, then you will have to copy the file under a different name so that you may use DU-V87 to make these changes we have been discussing.

Quite often when you are typing a lengthy article or document, you will pause to save the latest part that you have written. Not only does this help avert a loss of work but more than that, it is often desirable to actually see a hard copy. Actually reading a printout is the best way to evaluate what you have written.

This is all the more important, when you are using embedded printer commands. Did they do what you expected? It's not unusual to find that you forgot to turn off the italics or emphasized print. It's OK to stop every so often and print the whole thing. But frequently, you have proof read your text up to the place where you resumed writing and are satisfied with that part of the text. All you really need is to see the latest addition to your material. Why print out four or five pages of work you've already read? Well, here's where another feature of VDE comes in most conveniently.

There is a printing command that permits you to print part or all of your text while never leaving VDE. It prints from memory. You can double space the part you are proof reading but the text in memory remains unchanged. You can have as many copies (up to 255) as you wish.

To start the on-line printing you use the VDE print command, Control-K P (^KP). If you intend to print out just a block of text, you first mark the block. At the start of the block to be printed, place your cursor and hit Control-K B (^KB). Then move the cursor to the end of the block and hit Control-K K (^KK). After you hit the printing command (^KP) there will be a prompt in the header at the top of your screen for OPTIONS.

Here is a list of those options:

- B Prints only the currently marked block
- P Pauses for your keystroke before each page (sheet feed)
- D DOUBLE SPACING all the text to be printed
- Lnn Set the printer LEFT MARGIN to nn columns (The default can

(CONTINUED ON PAGE 14, COLUMN 1)

EXPLORING SmartBASIC

Part XIII: Playing with RAM

by Guy Cousineau

RAM stands for Random Access Memory. All of the 64K available under SmartBASIC is RAM area. This means that different values may be placed in any memory location, and modified, as required, by the controlling program. ROM (Read Only Memory) on the other hand, can only be read. One example of ROM are the game cartridges. The information on these cartridges is constant. The program contained in them is copied to RAM prior to program execution.

Before looking at commands that affect RAM, let's make a quick memory map of the standard RAM used by SmartBASIC.

ADDRESS		DESCRIPTION
00000-27407	(0000-6B0F)	SmartBASIC program
27407-??	(6B0F-??)	Variables stored up from here
??-54272	(??-D400)	User program stored down from here
54272-57344	(D400-E000)	3 1K buffers for directory and file access
57344-65535	(E000-FFFF)	Operating System

The space between the question marks (26865 bytes) is the RAM area available for user programs. As you write a new program, the top of available memory is adjusted down from 54272. When you RUN your programs, the space from 27407 gets filled upward for each variable and string that you define in your program. If the 2 ever meet, you get that nasty OUT OF MEMORY error.

FRE is used to find out how much memory is available. Since FRE is a variable command (similar to the trig function), it requires a parameter even though it is not used. Thus FRE(0) is the same as FRE(22). When you first boot SmartBASIC, you can ask for the amount of free space with:

```
PRINT FRE(0)
25954
```

But 25954 is less than the 26865 described above. That is because some variables are already defined (the variable commands) and take up a bit more than 900 bytes of RAM. The FRE command can be used within programs to check on the available space.

When FRE calculates the memory space, it begins by doing some house cleaning in the string space by crunching up strings which are no longer required. Then it reports the available memory. To force this house cleaning operation, add a line like:

```
f=FRE(0)
```

in a strategic location in your programs. This may help prevent the loss of valuable strings when sorting large string arrays.

PEEK is used to report the value in a particular memory location. This can be handy to check on the status of an operation, or to read data. Let's consider one application. I have a program which performs several LONG iteration (like GAME OF LIFE simulations). I want to give the user a chance to abort without having to press CONTROL-C. Since we know that the last keypress register is at 64885, we can PEEK that address to detect the abort request:

```
1000 GOSUB 2000: REM do one generation
1010 p=PEEK(64885)
1020 IF p=ASC("q") THEN END: REM quit request detected
1030 IF p=ASC("r") GOTO 100: REM restart request detected
1040 GOTO 1000: REM continue if no option detected
```

Since PEEK returns a decimal value, we must compare (in 1020 and 1030) with a decimal value by comparing to the ASCII value of the option letters. Note that we could have compared with 113 and 114 but ASC ("q") makes it clear that we are checking for the letter "q". When execution speed is not critical, this is a highly recommended programming technique.

POKE is the opposite of PEEK; it places a value in a RAM address. POKE 27407,25 for example would place the value 25 in location 27407. This can later be verified with PRINT PEEK (27407). POKEing below 27407 should be done with extreme caution as it will change the SmartBASIC program. POKEing above 54372 should also be done with great care since this will change the operating system. SmartBASIC protects against this by checking the POKE limit at 16145-16146 when the POKE command is executed. Novices should not change the POKE limit until they are ready to experiment with the E.O.S.

To disable the POKE limit: POKE 16145,255: POKE 16146,255. Alternately, you can POKE 10120,2: POKE 10121, 201 to defeat the check and make POKE run just a bit faster.

LOMEM can be used to reserve RAM area for user routines or DATA. LOMEM commands should be placed at the beginning of your programs since LOMEM also clears variable arrays before adjusting the memory pointers:

```
LOMEM: 28000
```

will set aside 593 bytes from 27407 to 27999 in which you can PEEK and POKE to your heart's content. You don't have to worry about overwriting your program, variables, or SmartBASIC itself. Note that the LOAD, CLEAR, and NEW commands do not affect the LOMEM setting. Programs that set LOMEM abnormally high may cause you OUT OF MEMORY problems later on when loading other programs. Should this ever happen, type NEW and follow with PRINT FRE(0). If you don't have enough memory to load in the new program, reset LOMEM: 27407 and LOAD again. As an added precaution, have programs which reset LOMEM return it to its normal setting as part of the exit routine.

HIMEM is similar to LOMEM but it protects RAM above the specified address. HIMEM is complicated to use since it requires knowledge of the size of your program. HIMEM must be located below the end of your program by an offset equal to the number of bytes you wish to reserve. As HIMEM has no apparent advantages over LOMEM, its use is discouraged. Contrary to LOMEM, HIMEM is reset every time a program exits.

The HIMEM execution will check that the HIMEM address is not overwriting the program or data segment of RAM and abort with OUT OF MEMORY. When it is successful, it saves the address at 16109 (3EED) which is the TOP address for numeric variables. It will not report how much RAM has been protected by this change.

CALL is used just like in MACHINE LANGUAGE routines to execute a routine at a particular address. This is an advanced command which should be left to experienced programmers. The CALL routine saves all the program pointers, executes the requested routine, restores the program pointers and clears the accumulator prior to resuming the SmartBASIC program. It is the programmer's responsibility to preserve the stack and/or set up a local stack.

USR is similar to CALL except that it always branches to the USR routine at memory address 16130-16131 (3F02). At BOOT, this address points to a RETURN and no harm can be done by a USR command.

USR is similar to CALL in that program pointers are preserved and the accumulator cleared on exit.

The advantage of the USR function is its ability of passing a parameter to the user routine. When the USR routine gets control, the DE register points to the function number with the high bit set. Since USR is a variable command, it requires a variable. Thus the correct syntax is:

$$a=USR(n)$$

where 'a' is any legal numerical variable and 'n' is a number from 0 to 255. Since the high bit is set, values from 128 to 255 are the same as 0 to 127. Following is a sample of preamble code to a series of user functions:

```
   LD    A,(DE)        ;get function
   AND   7FH           ;strip high bit
   CP    3             ;max function +1
   RET   NC            ;function out of range
   LD    HL,FTABLE     ;jump table for routines
   LD    B,0
   LD    C,A           ;function number to BC
   ADD   HL,BC
   ADD   HL,BC         ;point to function vector
   LD    A,(HL)
   INC   HL
   LD    H,(HL)
   LD    L,A           ;get execution vector in HL
   JP    (HL)          ;execute it
FTABLE:
   DW    FN_0
   DW    FN_1
   DW    FN_2
```

The & routine is similar to the USR function except that it does not preserve the DE register which is the pointer to current position in the command line. The routine is useful for those applications which will parse a series of instructions from the command line.

It is the programmer's responsibility to return the DE register pointing to next instruction and to clear error conditions in the accumulator. & gets its execution address from memory location 16132-16133 (3F04). At BOOT, this points to REM which effectively ignores the rest of the line.

You may often see programmers using & as a REM statement. This practice is not recommended for programs which will be distributed since other people may have installed & routines.

In order to use the & routine, it is necessary to know something about the register use in SmartBASIC. Although this will be covered later, this is the essential information:

Register DE contains the pointer to the current line being executed.

Register C in the alternate set has the number of characters remaining.

Following is a sample start up routine for an & function:

```
   EXX               ;use alternate set
   LD    A,C         ;get length of line
   LD    C,0         ;set line length to empty
   EXX               ;normal set
   LD    H,0
   LD    L,A         ;length to hl
   ADD   HL,DE       ;point de to end of line
   LD    (SAVEDE),HL ;save exit value for de
   INC   DE          ;skip header
   INC   DE          ;get length byte
   LD    A,(DE)
   INC   DE          ;skip to first character
   LD    B,A         ;save character count
                     ;
                     ;go on from here to decode the instr. in (DE)
                     ;using B as a down counter
                     ;
   ;all routines must exit to this routine to properly restore register DE
                     ;
EXIT:
   LD    DE,(SAVEDE)
   XOR   A           ;make a zero
   SCF               ;set the NO-ERROR condition
   RET
```

WAIT is another advanced command which is used to wait for a particular value in a port. Its use requires detailed knowledge of prot operations and status codes returned by peripherals. The syntax is:

WAIT a,b,c

where A is the PORT number, B is the XOR value, and C is the AND value.

The WAIT command will get the value from the port, XOR it with B, AND it with C and continue this operation until a non-zero result is obtained.

Using WAIT without the proper parameters will effectively lock up your system.

ROUTINE ADDRESS

- **FREE** executes at 10192 (27D0). It recovers unused string space, and subtracts the top of variables from the bottom of strings to determine the free RAM space.

- **PEEK** executes at 10091 (276B). It gets the memory address and loads its value into the floating point accumulator.

- **POKE** executes at 10104 (2778). It gets the memory address and checks it against the POKE limit. If the address is within range, it places the requested value in memory.

- **LOMEM** executes at 10870 (2A76). It gets the protected address and moves the variable tables accordingly. If insufficient memory, it aborts with a message.

- **HIMEM** executes at 11010 (2B02). It also gets the protected address and shifts tables accordingly.

- **CALL** executes at 10042 (273A). It gets the address, saves the registers, calls the routine, restores registers and return to the program.

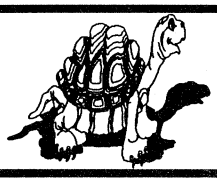
- **USR** executes at 10073 (2759). It saves the registers and extracts the routine address from memory. It then jumps to the CALL routine to finish off.

- **&** executes at 10164 (27B4). It saves some registers and extracts the routine address from memory for execution.

- **WAIT** executes at 10126 (278E). It extracts the PORT XOR AND, and cycles through until a non-zero result is obtained.

This concludes the analysis of RAM functions. Next time out, we'll cover STRING FUNCTIONS followed by Mathematical Functions and then Random Numbers in coming months.

ADAM NEWS NETWORK
c/o Guy Cousineau
1059 Hindley St.
Ottawa, Ontario, Canada K2B-5L9



THE LOGO NOTEBOOK

Part X & XI: A LOGO Tutorial from Scratch

by Ron H. Mitchell

We now know a thing or two about the LOGO language. Let's see if we can put our knowledge to work. This may take a while. Over the next few editions of the notebook, we're going to begin building our tutorial.

First, some planning. It should be pointed, I suppose, that this is not the only means or method of going about the production of a software program, but it might inject some order into the process.

PURPOSE

Question:
What do we want this program to do?

Answer:
To serve as a 'notebook' on the LOGO language. To display the various LOGO primitives, to describe their purpose, provide an example of their syntax, and to suggest one or more examples of useage in each case.

Question:
For whom is the program intended?

Answer:
For the novice LOGO user who is setting out to learn more of the language.

GENERAL CONCEPTS

The program will be menu driven and will permit the following functions:

- Viewing of each LOGO primitive as an entry on a notebook page supported by accompanying syntax and examples of useage.
- Viewing of information either in sequence or by primitive - ie; selective access to commands and operations
- Addition, editing and deletion of user written procedures.
- The entry of notes by the user to provide reminders of various aspects of the command, operation or procedure under consideration.

We'll leave it at that for the moment. We don't want to get too complicated. The basic operation will be fairly straight forward; present the menu, get the user's choice, branch to it and display the selected LOGO primitive along with related notes and instructional material.

So... let's get started.

First, the main procedure. We'll take a preliminary run at it, even though it will no doubt change later on. What do we want it to do?

For the moment we'll look to our main procedure to set up the opening screen, get the user's selection, and branch to the appropriate procedure controlling each of the various sub functions. We also might want to include some elementary checking to prevent things from running off track if the user presses the wrong key. All of this will be reasonably simple.

The sub procedures will require a little more thought. We have essentially four functions that we want to perform: VIEW, ADD, DELETE and EDIT. (Can you think of any more?) These are the words that will be displayed on our opening menu along with QUIT, LOAD, and SAVE. This will permit us to load in new procedures to our tutorial that we've devised ourselves, and to save other procedures for future use. It will also allow us to leave the program gracefully when we so wish.

We also have to decide on a screen layout for the information to be presented, and on the best or most efficient means of arranging our information in ADAM's memory and on ADAM's tape / disk drive. In fact, there's a fair amount to be considered. There may even be things we'd like to see in the program that we haven't thought about yet.

It seems like a good idea to stop right here, and give you until next issue to put your own thought into this excercise. See if you can come up with a LOGO 'mainline' that will do the job described above. In the meantime, I'm open to suggestions about the program we're collectively working on.

I'll present my own version next article.

THE LOGO NOTEBOOK

Part XI: The LOGO Tutorial

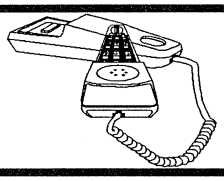
by Ron H. Mitchell

So how did you make out? Any progress on your version of the dreaded LOGO Tutorial?. Last issue we outlined in very basic fashion a tutorial program, and stated what the program was to do.

In this issue, we'll set up the opening screen, or the first part of it at least. The program is as follows; type it in and try it out.

```
TO KEY1
  PU
  SETX -125
  SETY -90
  SETPC 5
  KS
  SETCURSOR [2 21] TYPE [I]
  SETCURSOR [1 22] TYPE [ADD\ ]
END

TO KEY2
  PU
  SETX -85
  SETY -90
  SETPC 4
  KS
  SETCURSOR [6 21] TYPE [II]
  SETCURSOR [5 22] TYPE [DEL]
END

TO KEY3
  PU
  SETX -45
  SETY -90
  SETPC 5
  KS
  SETCURSOR [11 21] TYPE [II]
  SETCURSOR [10 22] TYPE [EDIT]
END

TO KEY4
  SETX -5
  SETY -90
  SETPC 4
  KS
  SETCURSOR [16 21] TYPE [IV]
  SETCURSOR [15 22] TYPE [VIEW]
END

TO KEY5
  SETX 350
  SETY -90
  SETPC 5
  KS
  SETCURSOR [22 21] TYPE [V]
  SETCURSOR [20 22] TYPE [FILE]
END

TO KEY6
  SETX 750
  SETY -90
  SETPC 4
  KS
  SETCURSOR [26 21] TYPE [VI]
  SETCURSOR [25 22] TYPE [QUIT]
END

TO KS
  SETH 0 PD FD 20 RT 90 FD 40 RT 90 FD 20
    RT 90 FD 40 RT 90 FD 20
  PU
  SETH 120 FD 10 PD FILL PU
END

TO BORDER :T
  MAKE "J 1
  CS
  SETPC 1
```

(CONTINUED ON PAGE 14, COLUMN 2)



OVER THE PHONE LINES

Part II: Lurking is Permitted for Beginners

by Bart "Zonker" Lynch

Well, seems as though it's time to sit down and write part 2 of this series. I forgot to include my address in part 1, so that you could write with comments!

My address is:

Bart Lynch
4511 S. 256th
Kent, WA 98032

Please feel free to write with any questions, criticizms or (perish the thought!) praise. Your input is appreciated as well as a Self-Addressed Stamped Envelope.

If this is your first time on a BBS, it would be a good idea to leave the SYSOP some feedback, telling him so. Then, if you make a few "mistakes", he'll know why! And don't be afraid to ask for the sysop's help. Part of the reason a sysop puts up a BBS is to help others out. This is particularly true of ADAM BBS sysops. They remember the first time THEY went on-line, and want to help you so you don't make the same mistakes THEY did!

On your first visit, you may be a little overwhelmed by the menus and the commands. Even if you make it to the messages bases, it can all seem a little confusing. Don't worry about jumping in and participating straight away.

Spend the first few visits getting familiar with the setup. It might also take some time to get into the "flow" of conversations taking place. Soon enough, you'll be in the thick of things. For now, it's best to just observe.

I realize by saying that I might be putting myself in hot water with the sysops! In the BBS world, there are few things worse than a "lurker". What is a lurker? Someone who logs on and just reads the messages and does not join the conversations himself!

But it certainly isn't out of the ordinary to lay low your first couple of visits.

When it comes time for you to "speak out", leave a general message introducing yourself. Leaving a message on a BBS is called POSTING and usually guarantees several RESPONSES. Heck, let the other users know you're new to the BBS world. You'll find them helpful too in navigating your way through a BBS.

Always keep in mind that you are speaking publicly when on a BBS. This means (that you should use) common coutesy. Never say anything in the public area of a BBS that you wouldn't say to a person's face. (Or in front of your mother, for that matter!).

Sometimes it's difficult to judge the contents of a BBS message because of the fact that it's written, not spoken. I've known a few arguments to break out because of misunderstandings, and things taken out of contact. Even between such well-mannered folks as ADAMites! Try to read carefully what the other person is trying to say and not jump in half-cocked.

Some of the BBS "talk" can get kind of technical. Don't let this scare you off! If you have nothing to contribute to that particular discussion, start one of your own. I know that Barry Wilson is on several BBS' and Barry isn't exactly what you'd call a technical person! And yet sysops all over welcome him. Why? Because he's not afraid to make comments or ask questions. This, above all, is the heart of a BBS. Input is vital or there would be no messages to read!

I know that some folks would have you believe that another reason to use your modem is to get to those "free" public domain programs. I say "free" as I have never come across such a critter! When you take into account your on-line time (and the money your local phone carrier charges for that time), you end up paying someone for those supposedly "free" programs. However, it IS a CHEAP source of programs and they are usually on a BBS before some buying service or your local user group has them available.

Unfortunately, describing HOW you get to those programs is beyond the scope of this article. For that information, you'll have to rely on the sysop or BBS menus. Each system if different, in that regard. But, once you get there, downloading is a snap! As I said in part 1, I'm assuming you have at least ADAMLink II when I give the following instructions.

When you've selected the file you just can't live without and chosen to download the file, follow these steps:

1) Press the WILDCARD key. This takes you to command mode. Don't worry! You're still logged on to the BBS. You've just reverted the screen to your ADAM output.

2) Next, press SmartKEY VI (labeled FILE).

3) Next, press SmartKEY II (RECEIVE).

4) You'll then be prompted for a drive to download the file to. Make sure the drive you choose contains a formatted data pack or disk which you know has enough space free for the entire file you will be downloading.

5) Next, you are prompted to enter a filename. This can be ANY name you wish (it does NOT have ato match the filename on the BBS!) up to 10 characters long.

6) Next, press SmartKEY VI (DONE).

Now you're ready to download!

7) Just press the WILDCARD key again to get back to the BBS.

8) To SAVE your file after it's been transmitted, merely press the UNDO key. This will write your file to the media and you're all set!

That looks like as good a place as any to stop for this time. Next time, we'll look at bulletin boards that are ADAM specific. Until then, good luck!

CHICAGO SYSLINK

Chicagoland's Window to the Fidonet ADAM Echo

SYSOP: George Matyaszek
FIDONET #: 1:115/622
PHONE #: (708) 795-4442 - 300-2400bps
PARAMETERS: 8-N-1
HOURS: 24 hours a day, 7 days a week

Call and leave George a message requesting information on how to join the ADAM Echo and tell him N.I.A.D. referred you.

MOVE & COPY TUTORIAL, CONTINUED FROM PAGE 6

You might want to add texture or shading with Brushes. You might want to REVERSE SCREEN to transform foreground into background and vice versa, without affecting any of the rest of the picture. You might want to try any number of experimental changes before making them permanent. You might want to split a clip or sprite so that you can place part of it in one cell and part in another. All those options are open to you from the MOVE/COPY menu. Don't be afraid to experiment. Have fun!

BEYOND SmartWRITER, CONTINUED FROM PAGE 7

by the program, and an all-keyboard, Control + key system, means your fingers do the work without conscious thought, or by reading the SmartKEYs, after you learn the program.

VDE offers most of the bread and butter features you expect in any word processor, but with the additional feature of Macros. Macros are really little programs that you can write into your program to automate many regular functions you may otherwise have to repeat.

I am not a major Macro maven, (but simply an annoying alliteration addict,) but when you progress to Macros, you can teach your greyhound some tricks Benji never heard of.

VDE's 43K file size limit is occasionally referred to as a drawback, but that's approximately twice the file size of SpeedyWRITE and SmartWRITER.

It has another companion program, enigmatically titled CHOP.COM, that can readily separate the files of your next novel, into bite-sized chunks, anytime you want to work on it in VDE.

VDE266.COM is available in an ADAM-specific version (ADAM266.VDE) that works with the ADAM keyboard and printer.

All of the arrow key functions you're used to, insert and delete keys, are still available and functional, (but no backspace unless you specifically set it up with VINSTALL.)

VDE is in the public domain -- i.e. it is a totally free program that's in the VISA P.D. library, available from ADAM's House, or N.I.A.D., for their P.D. prices, and should be in everyone's ADAM disk or data pack collection.

VDE does run from and store to data packs, but with the proviso that it must run under T-DOS or CP/M and those programs don't run well from data pack.

With the money you save on VDE, you can send away for a disk drive!

To wrap up: VDE is easy to be enthusiastic about. It's only real drawback is it's operating system - if you are new to TDOS or CP/M, you'll have to learn two program. On the other hand, the benefits are real, and the effort is rewarding; you get a very competent, feature filled word processor that expands the usefulness of your ADAM, and at an unbeatable price.

PUT IT IN WRITING!, CONTINUED FROM PAGE 9

be set with VINSTALL)

- ● ^ FILTERS out the job nn times
- ● *nn Prints out the job nn times
- ● @nn Begins printing at nn page
- ● #nn Prints only up to a TOTAL of nn pages
- ● "..." Uses the quoted string as a HEADER. The string, followed immediately by the page number, will print at the top right of each page, near the margin. (Maximum length of the string is 50 characters; and empty string "", gives numbered pages with no header text).

You may use as many of these features at the same time as you wish. For instance, if you choose BD you will get a marked block printed in double spacing. The filter command will print out the control characters in quotation marks. ^K would print as "^K". I highly recommend the printing feature to you. One more advantage that comes to mind is that you can check the size of your copy. Suppose you are nearing the end of an article (as I am here) and you must use care not to exceed that allocated space. You can see the actual printout and just how much space you have to work with. This on-line printing lets you keep a very close tab on the way your space is being used up. It is a powerful editing aid as well. If you have seriously reworded a paragraph or two right in the middle of an article, or if you have inserted a section from another file (VDE will let you do that) you can just print out that part of your text to see how it turned out. All this while VDE still has your work in memory for you to go back and adjust anything that doesn't look or read correctly.

I have just mentioned the manipulation of a block of data. There are a number is very useful block control commands but it is time to close out this session on the use of VDE.

To be continued.

THE LOGO NOTEBOOK, CONTINUED FROM PAGE 12

```
  PU SETPOS [-120 90] RT 90 PD FD 240 RT
    90 FD 160 RT 90 FD 240 RT 90 FD 160 PU
  MAKE "X -120 + :T MAKE "Y 90 - :T SETX :X
    SETY :Y PD RT 90 FD 240 - :T *2
  RT 90 FD 160 - :T *2 RT 90 FD 240 - :T *2 RT 90 FD 160 - :T *2 PU
  SETPOS [-114 89] PD FILL
  PENUP HOME
  SETCURSOR [3 3]
END

TO KEY
  KEY1
  KEY2
  KEY3
  KEY4
  KEY5
  KEY6
  PU
  SETH 0
  SETPOS [-115 -90] RT 90 PD FD 235 PU SETPOS [-115 -70] PD FD
    235 PU
  SETCURSOR [2 3]
  CHANGE.COLOR 15 1
END

TO SETUP
  SETBG 7
  BORDER 5
  KEY
  SETCURSOR [1 2]
END
```

What would you do next?

(ED. NOTE): Please direct all questions and suggestions to the author of this series and include a Self-Addressed Stamped Envelope if you wish to receive a reply. While I have a little experience with SmartLOGO programming, I am rather rusty on the subject matter and would not be able to supply a knowledgeable and also prompt reply. As the old saying goes, "Ask the expert!".

THE ADAM-USER FRIENDLY GROUP
c/o Ron H. Mitchell
Apt. 1107, 210 Gloucester St.
Ottawa, Ontario, Canada K2P 2K4

TWEAKING T-DOS

Part II: A few notes on getting the most power from T-DOS

by Ronald Collins

USING XSUB TO AUTOMATE DIFFICULT OR REPETITIVE TASKS

This next section was taught to me by Guy Cousineau at ADAMCON 03. Guy and I had been discussing some of the things he had figured out as real time savers. Some of what we've covered above was first learned from Guy at that convention. Using XSUB is one thing I've never quite understood. What's it for and what does a person use it for? Well, thanks to Guy, I now know some of what XSUB can do!

"XSUB" is a program that can be used with a submit file to "fill in the blanks" so to speak. If you've ever used an installation program, you know how repetitive it can be to copy each command over and over again. PASCAL programs, for instance, all use the same installation routines. Rather than repeat the same answers to each question you are asked by the installation program, why not do it only once?

The trick is to write down your answers on a piece of paper as you type them in. Your response to each question, in the same order you made it, can then be written into a submit file with a word processor for later use with other programs. When you want to install another PASCAL program for use on your ADAM, just use your new INSTALL.SUB to do it for you automatically!

What started Guy's lesson was my mention that I sometimes reinstalled T-DOS for different configurations, depending on what I happened to be working on. I might need 40 column or 80 column video, need to support only data drives or maybe 4 disk drives. I might only need to install a new version of T-DOS on my normal hardware set-up. In each case, a small submit file could be made to answer all the T-DOS installation questions as they are asked.

At Guy's request, I went through a complete T-DOS install, right down to installing SmartKey strings. As each question was answered, I copied the answer to it onto a piece of paper. When this was all completed, the paper was transcribed into a SUBMIT file using ZDE. XSUB was used to send each answer as needed. It all ended up looking like this:

INSTALL.SUB FILE

```
e15:xsub.com
80bhd45.com
3
b
3
5
0
2
15
0
1
3
0
y
a0: e15: a1: c0:
y
y
y
n
n
n
```

Each of the blank lines you see is simply a <RETURN>. (ED. NOTE: There should be one blank line in between each line of the SUB file listed above but due to space limitations we have not included them).

The use of the submit file is pretty easy. Just type INSTALL 80TDOS and press <RETURN>. T-DOS will read the file named INSTALL.SUB, follow it's first command and load XSUB.COM from drive E15:, and then load the program 80TDOS.COM. As 80TDOS asks for information on how I want it to be set up, the answer will be read by XSUB from the submit file. The answers will be supplied in the order they are written as in a SmartBASIC data statement!

When it's all done, I can simply pull the computer reset to load up my newly installed copy of T-DOS.

Another, simpler, use of the XSUB command in a submit file can be demonstrated. I have a very powerful database package called CONDOR. This package has certain questions that must be answered each time you boot it. As the program is loaded, it asks you for your license number. You must fill this in if you want to edit data fields, etc. Next, it wants to know what your master disk will be. This is where CONDOR will look for all of it's support files. Finally, to look proper on your video screen, you must tell it what terminal emulation is being used. That license number is nobody's business but your own (to stop software pirates), but the rest is essential to the use of CONDOR. For this reason, the license number can not even be used with XSUB... it just ignores the submit file data until the user either supplies that number manually or presses the <RETURN> key. Once the return is pressed, CONDOR knows that someone else is using it and limits some of the program capabilities. This is useful for employees to enter data while you keep the ability to edit parameters. As soon as it's been directed by this info, the software will then except data supplied by the XSUB file.

I keep CONDOR on C12:, so drive C: will be my MASTER directory. To tell this to CONDOR, I have to type SET MASTER C: and press the return key. To tell it that my terminal type resembles a LEAR ADM-3, I have to type TERM=LEAR and press the return. The submit file then is written as:

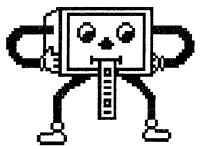
CONDOR.SUB FILE

```
E15:XSUB.COM
C12:CONDOR.COM
SET MASTER C:
TERM=LEAR
```

To use it, I've named the submit file, CONDOR.SUB. When I type CONDOR and press the return, the submit file is loaded and a quick booting is taken care of.

I'm sure there are many similar uses that you can find for combining XSUB and the built-in submit processor. If you've found any other good ideas on streamlining the usage of T-DOS or programs under T-DOS, let me know. I think a sharing of information is essential to keeping the ADAM community more of a FAMILY. If nothing else, I've learned a long term need for attending those ADAM conventions each year.

I hope to see a good turn out at the 1992 convention - July 23 to the 26th of 1992.



TDOS, SMARTKEYS AND YOU

SmartKEYs make life easier when used in TDOS

by Bob Blair

Anyone who uses CP/M or TDOS knows that when you boot up those operating systems a listing of SmartKEY commands appear at the bottom of your screen. When you press a SmartKEY (the black keys), a command is typed on the screen which is the executed by pressing RETURN. The commands include things like: DIR - for directory of files on disk / ddp, or DEL - to delete a file, etc. The SmartKEYs just type the name of frequently used commands on the screen for you as a time saver.

TDOS, however, gives you the ability to decide what command name the SmartKEYs will print on the screen, and can be set to execute the command for you, as follows:

When you originally set up TDOS on your data pack or disk, one of the many questions you are asked is: Do you want to "change the SmartKEYs?" If you type in "Y", for yes, you are then prompted to rename each of the SmartKEYs. But did you know that in addition to having the SmartKEYs type the name of a command on the screen, you can also have them automatically execute that command without you having to press the RETURN key? It's easy!

When you tell TDOS what Command you want a SmartKEY to print on the screen, put a **^M** immediately following the command name. For instance, if you frequently use a program called MAINT12.COM (by Guy Cousineau), type in:

MAINT12^M

when answering the above question for SmartKEY I. Thereafter, when using TDOS, to execute this fantastic utility program called MAINT12, just hit SmartKEY I, and it types the command name on the screen and (in effect) presses the RETURN key for you.

The above feature, plus the ability of TDOS to automatically perform a whole series of commands and tasks through the use of ".SUB" (short for SUBMIT) files, sets the stage for the introduction of a very useful new series of programs from Guy Cousineau. These new programs allow you to execute a single command, while using TDOS, which changes your SmartKEYs to almost anything you want. And it includes the ability to execute another command to reset those SmartKEYs to the original settings.

But, what can this do for me???

Imagine this:

By pressing a single SmartKEY, you could. . .

1) Boot up the TDOS installation file, have all the questions answered for you and have the TDOS system written on a new data pack or disk; or,

2) Boot up the word processor called VDE266.COM and automatically have a whole new set of SmartKEYs available at the bottom of the screen which would perform things like FORMAT, SAVE, PRINT, QUIT, TOP, BOTTOM (of page or document). And when you do QUIT, the original SmartKEY settings appear at the bottom of the screen; or,

3) Execute your favorite modem program (plus dial the number, using MEX) and have a new set of SmartKEYs with commands for your modem operation like: YOUR NAME, PASSWORD, DOWNLOAD, SAVE, UPLOAD, EXIT. And after you saved the messages from the BBS to a file and exit from the modem program, the file you saved is called up in the VDE266 program automatically, complete with new SmartKEY settings to enable you to automatically go to any message containing your name (FIND your name), NEXT PAGE, PREV PAGE, TOP FILE, BOTTOM (of file) and SAVE. When you exit VDE266, the original SmartKEY settings are reset and the file is sent to a data pack or disk after being "crunched" (all blank spaces removed in text) for archive purposes, and the screen cleared.

And I'll bet you can invent ways to use the SmartKEY setting programs beyond my above uses.

If you are a M.O.A.U.G. or 463 ADAM member and want to try out the new SmartKEY setting programs yourself, send me a formatted blank data pack or disk, and sufficient postage for me to mail it back to you in the same package, and I will send you these neat programs, including ".DOC" files explaining how to operate them (ED. NOTE: The T-DOS V4.59 volume in the N.I.A.D. Public Domain Library contains these SmartKEY programs or you can follow Bob's instructions on how to obtain them). In fact, if you include a note that tells me about the drives, expander and printer you are using (like: one 160K Disk Drive, two Digital Data Drives, 64K Memory Expander and ADAM Printer), I will pre-set the data pack or disk to boot up with the SmartKEYs set for you, and include a few surprise programs that I designed which you may find useful.

Also, if you let me know what programs you use most often, I will include those in the SmartKEY settings. Besides, it will be interesting to me to find out what programs T-DOS users find most useful.

A special THANKS to Guy Cousineau for writing the SmartKEY settings programs and to James Poulin for including them on his "Wish List" for Guy to write, and then sharing then with us.

SOUTH FLORIDA A.U.G.
c/o Bob Blair
361 Eden Dr.
Englewood, FL 34223

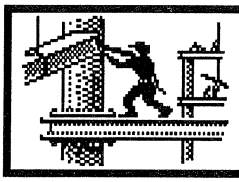
THE A.N.N. WISH LIST

The ADAM News Network Wish List is the brain child of Guy Cousineau, a prominient member of the A.N.N. Staff as well as the ADAM community and also one of the most talented and prolific programmers of ADAM E.O.S. and TDOS software.

If you have any ideas, wants or needs when it comes to software programs, patches, enhancements, etc. or just about anything else related to the ADAM Computer, write Guy a letter detailing in full what you have in mind.

If it is within the realm of possibilities and will be of benefit to others, you can rest assure that your suggestion(s) will be looked into very seriously (as can be attested to by the creation of the SmartKEY programs for TDOS that James Poulin suggested).

A.N.N. WISH LIST
c/o Guy Cousineau
1059 Hindley St.
Ottawa, Ontario, Canada K2B-5L9



TOOL TIMES WITH THE ADAM

Printer Fan Installation

by Guy Cousineau

Some time ago, I asked David Cobley (VISA) about installing a cooling fan in my printer. He graciously sent me a copy of an article from Bob Stroud (Calgary) which contained detailed instructions. David added a few comments of his own, including powering the fan directly from the Adam Printer rather than using an external source. I found that I had to be a bit more creative than initially anticipated, but the whole job took about two hours. Now my printer never seems to overheat. For the benefit of those who are contemplating such a task, following is a rehash of the instructions.

PARTS REQUIRED

- 1 110 volt low impedance cooling fan (up to 3 1/2 inches)
- 2 stove bolts or tapping screws big enough to fit snugly in the holes
- 1 finger guard (optional)... see the details below
- 1 Phillips screwdriver
- 1 paper and pencil to take notes

Start by unplugging the printer and moving it to a comfortable working area. Turn it upside down and remove the deep inset screws at the edge of the printer. These are the ones that hold the cover in place. Carefully turn the printer over and remove the upper shell. Watch for the sleeve that fits over the paper tension guide; be sure it does not fall inside.

Your fan will be installed on the left side at the back of the printer. Not only is there sufficient room there, but it is the main source of heat. Cut away the pedestal on the left side (not the back) to make it level with the rest of the casing. Although a bit smelly, I used an electric soldering iron to melt it down; it was less trouble than trying to reach in with a file. Next remove the screw that holds the corresponding pedestal in the upper shell. The ridges in the shell must also be smoothed down to ensure a proper fit.

The next step is to find power for your fan. This entails the removal of the power supply and logic board: not for the faint hearted. You will find two screws at the back of the printer near the spot where the cords come in. Near the center of the printer (underneath the print head), you will find yet another mounting screw. On the under side of the printer, there are four more screws. These are the ones nearest the centre of the printer. Mine were mounted on a metal bar (for added sturdiness?) It is recommended to tilt the printer slightly during this process rather than turning it over.

Now look at the front of the logic board. The part to the right and under the print head. There is a connector on the extreme right edge and a strip of connectors to the left. Take note of the colors and number of wires in each connector and gently pull each one out. On top of the Heat sink at the back, you will likely find a ground wire running from the solenoid that advances the platen. Remove the screw and push the wire out of the way. Working form the back of the printer, lift the power supply slightly to clear the lip and begin edging it out. Watch that none of the wires or components get caught anywhere and pull gently. If it seems stuck, have a good look around. Perhaps you forgot to remove one screw ?

Turn the power supply over to reveal a metal shielding over the power leads. Remove this shield and examine the wires. You will see that one of the wires from the POWER source goes directly to a connector and that the other is re-routed to the switch. Use the return wire from the switch and the other main power wire as your source for fan power. Strip a short bit of wire from the fan and solder it over the existing wires. Be sure to get a good connection. Then channel the wires towards the edge of the card and tie them down.

Although you can use electric tape, I prefer HOT GLUE as it gives a more secure bond. Before proceeding, you may wish to verify that the fan indeed has power. Check to make sure you have no possible short circuits and plug the printer back in. Turn on the power and check the fan. Be sure to UNPLUG the printer afterwards.

Replace the shielding and re-insert the power supply back in the printer. Re-insertion is trickier than removal. You must watch out for the cable that runs to the solenoid at the back. Once the board is in place, make sure that the said cable does not interfere with the print head movement. Try sliding it back and forth to make sure there are no obstructions. Replace the 7 screws holding the power supply. Insert each screw only part way until they are all in. This will allow minor adjustments to line up the holes. Be sure to tighten everything back up after. Then re-insert the connectors on the logic board... IN THE RIGHT SEQUENCE!

At this stage, you may wish to verify that the printer appears to be working correctly by plugging it in again. Make sure the fan is not in the way before doing so. If it whirrs as it normally does on power up, you are probably ok. While you are there, you may wish to clean the knife switch on the left side of the front of the printer.

The next step is to decide on the exact location of the fan. I centered mine between the two ridges on the casing. I then made a template form a piece of cardboard with the four holes from the fan. With the fan mounted in place, I lined up the top two holes and marked the location of the bottom two. I drilled holes in the bottom of the casing, of the size of my screws. Because the printer cover is angled, I wanted to recess the fan a bit inside, so I used two large rubber faucet washers as spacers.

OPTION ONE

I then attached the fan to the bottom part of the casing. Then I placed the cover over the printer and, using the template, I marked the outline of the fan on the upper part of the casing. I then made a jig for my router and used a 1/8 bit to trace HORIZONTAL GROOVES in the casing. Although the lines were not perfectly straight, I though it more frugal and neater than paying another two bucks for a finger guard. I then placed the cover back on again and refastened the whole thing together. I had two screws leftover: one that held the inner pedestal, and one that attached the cover to said pedestal. If you have any more leftover, look things over again!

OPTION TWO

If you prefer to use a finger guard to the messy router approach, use the finger guard as a template to mark your first two holes on the bottom. Once those are drilled, replace the cover TIGHTLY and use the finger guard again to mark the top two holes. Using the finger guard, you can probably easily locate the center of the HOLE and mark it. You can then use a compass, or the finger guard itself (if it has round rings) to outline the circumference of the hole. Then make your cutout: be sure not to make it too large or you will break off the mounting holes to hold the whole thing together.

Although you may wish to try with a HOLE CUTTER, I prefer the coping saw approach, followed by a round file for trimming. Then, mount the finger guard on the outside and place the bottom two screws. Replace the top shell of the printer, being careful to sandwich it properly between the fan and the finger guard. Attach the top two screws and the bottom screws (now seven) which hold the upper shell in place.

PRODUCT REVIEWS

TYPE
GRAPHICS UTILITY

RELEASED
SEPTEMBER 1992

SmartPRINT PLUS

By The Maine ADAM Library

Reviewed by Bob Slopsema

PRICE
$15.95

MEDIA
DISK OR DDP

Have you ever had the desire to dress up your latest creation with some really unusual and attractive graphic clip-art or PowerPAINT screen designs? First off, you need a program that will load any graphics type we have available for the ADAM. Next, you would really like to be able to enlarge a portion of the graphic or even the entire graphic picture. But what you really want to accomplish is to turn the picture 90 degrees to one side or the other, or maybe even upside down!

If you have been looking for a program to accomplish all of this in one package, then SmartPRINT + from Maine ADAM Library is it! This program can accomplish the same routines that are in DEI's SwiftPRINT and DEI's SimplePAINTER - that of enlarging and flipping a picture upside down - plus it can turn pictures 90 degrees to the side with just the push of a SmartKEY! You can even turn 270 degrees using a combination of flip features together.

All documentation for SmartPRINT PLUS is contained in a number of SmartWRITER "DOC" files which you should print out and thoroughly read over before booting the program. Then, boot the disk - that is of course, AFTER you have made a backup copy! Now you are treated to background music while the program loads (classic Coleco type program). A SmartKEY screen much like SmartWRITER will appear whereupon you have six choices: loading/saving, moving/enlarging, flipping the picture, printing the picture, changing foreground and background colors, or exiting to SmartBASIC.

After the selection to load/save a picture, you are given the additional options of changing drives and a chance to catalog that drive. Even the ramdrive option is included, provided you have set up the ramdrive before running SmartPRINT +. You may load up to six different forms of graphics - "IBM" clip-art, PowerPAINT, HGR, SmartPAINT, RLE and ADAM clip art. Saving the adjusted/reformatted picture can be accomplished in only three formats - SmartPAINT, HGR or RLE.

You have the option to move the picture around on the screen before saving, or enlarge 1/4 of the picture to cover the entire screen.

You may also flip the picture upside down or 90 degrees, or a combination of both.

Printing pictures or clips presents two options. One is to use draft (with the printer supplying darker "letter quality printing" using its own built-in command panel). The second is to use expanded printing. This option opens up an wide array of choices. You can print your picture from 1" to 8" wide, and from 2" to 9" high, conceivably one picture on-screen could fill up the entire sheet of paper. This option also gives you a choice to choose the left margin when printing less than 8" wide.

Background color and also foreground colors may be changed to any of ADAM's 15 available colors.

And finally, you are given the option to leave the program and go to SmartBASIC without re-booting the computer.

A couple of problems I encountered while using the program. First, name length when saving a picture is very important -too many letters and the extension is overwritten making the picture inaccessible. My fault though, as the instructions point this out. Second, the HGR pictures saved are not accessible from PowerPAINT, but are accessible from SimplePAINTER, a problem I believe to be inherent in the PowerPAINT program itself.

This is not a new version or look-alike program to PowerPAINT or any other graphics program, but is a graphics printing and manipulation program. Very comprehensive and detailed instructions are included as README files, accessible from SmartWRITER. No memory expander is required to run the basic program, but the ramdrive will access the memory expander if available. A dot matrix printer IS required for printing.

This program is geared toward the novice ADAM owner as well as the advanced graphic's enthusiast with its SmartKEY menu's and broad variety of accessible graphics. And best of all, the price is right, $15.95 including shipping!

This program comes to us from the Maine ADAM Library and has taken a lot of thought and testing input to reach the final product. Everytime an updated version was finished, the program took on a new and better form - until we have todays final package, version 2.40. That is, of course, unless Bob finds a few more little goodies to squeeze into it. Order from the:

THE MAINE ADAM LIBRARY
c/o Bob Sebelist
P.O. Box 85 - Grover Rd.
Waterford, ME 04088

(ED. NOTE): The Software and Overall Ratings where note supplied by the reviewer, rather they were assigned by myself. A follow up review will be published in an upcoming issue for those of you who are interested. Also note that SmartPRINT PLUS is a direct modification of the SwiftPRINT program released by Soloman Swift of Digital Express Inc. in 1988. Unlike SwiftPRINT, SmartPRINT PLUS does not require a Memory Expander to work, but it only offers a one Cell workspace and does not support color printing and shading. Complete details and comparisons will be supplied in the review.

SYSTEM REQUIREMENTS

- **MEMORY:** BASE ADAM SYSTEM WITH R80 REVISION
- **PRINTER:** DOT MATRIX PRINTER
- **DRIVES:** 1 OR MORE DISK DRIVES AND OR DIGITAL DATA DRIVES
- **OTHERS:** NO OTHER REQUIREMENTS
- **OPTIONS:** PowerPAINT & COMPATIBLE PROGRAMS, MEMORY EXP.

SOFTWARE RATINGS

- **GRAPHICS / SCREENS** ..... A-
- **MUSIC / SOUND** C
- **INSTRUCTIONS** ..... B
- **EASE OF USE** ..... A-
- **VALUE FOR THE DOLLAR** ..... A-

OVERALL

B+

NIAD Northern / Illiana ADAM User's Group
8125 W. Catherine, Chicago, IL. 60656
Phone #: (312) 775-0409

MULTI - FUNCTION USER GROUP
MONTHLY NEWSLETTER - PUBLIC DOMAIN LIBRARY
DISCOUNT BUYING SERVICE FOR HDW & SFTW

Please DO NOT write in this space
PR 1 2 3 D/R

BUYING SERVICE ORDER FORM

Your MEMBER # _____

Name	*Name*
Address	*Address*
Phone ( )	**(Shipping address - If different)**

Qty.	Item Name / Number	DDP / Disk	Price Each	TOTAL PRICE

Sub Total

Please do not forget SHIPPING/HANDLING. Use the chart Below

Sub Total Range		( ) US UPS	( ) US MAIL	( ) CANADA
20.01	50.00	$ 5.00	$ 5.00	$ 5.00
50.01	100.00	$ 6.00	$ 6.00	$ 7.00
100.01	200.00	$ 7.00	$ 7.00	$ 8.00
200.01	and up	$ 8.00	$ 9.00	$ 10.00
		$ 10.00	$ 12.00	$ 14.00

For UPS C.O.D. orders... add $4.50 to the Shipping & Handling Charge. Cash terms only accepted. NO CHECKS!!

Ship / Hndl. _____

Initial ANNUAL Membership Fee required for Monthly Newsletter

New or Renewal Fee _____

PAYMENT METHOD

☐ **Check or Money Order Enclosed**

Charge my: ☐ VISA ☐ Master Charge

$25 minimum charge

Write Credit Card Number in Boxes

Exp. date: _____

Signature: _____

Balance or Credit Due _____

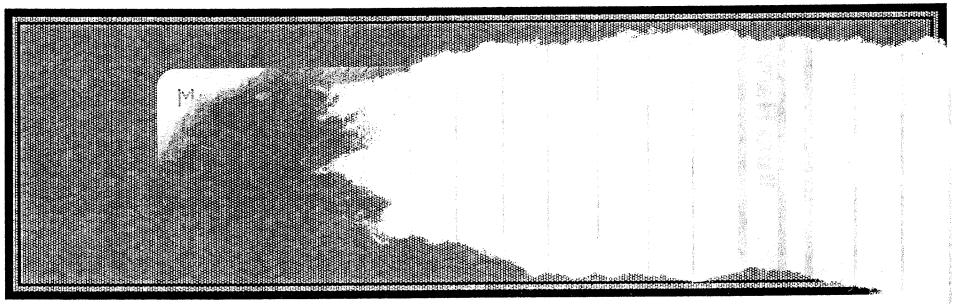
U.S. Funds Only

TOTAL INCLUDED _____

NIAD Northern / Iliana ADAM User's Group
8125 W. Catherine, Chicago, IL. 60656
Phone #: (312) 775-0409

MULTI - FUNCTION USER GROUP
MONTHLY NEWSLETTER - PUBLIC DOMAIN LIBRARY
DISCOUNT BUYING SERVICE FOR HDW & SFTW

N.I.A.D. NEWSLETTER FOR ⟶



ADDRESS CORRECTION REQUESTED

POSTMASTER: DATED MATERIAL - PLEASE DELIVER