

North Texas PC Users Group

8.10

October 1989



North Texas PC NEWS
(STARMAIL ADDRESS 51563)

Published monthly by members of North Texas Personal Computer Users Group for their use. Members each receive a free subscription; for others, price of the NEWS is \$2 per copy. Members are requested to notify the Membership Director in writing of address changes. Send all editorial correspondence to: Editor, North Texas PC NEWS, 4302 Ranier St., #111, Irving, TX 75062.

Publisher
John Pribyl (817)275-4109
STARTEXT MAIL CODE 51563

Editor
Doug McQuaid (214)255-1732

Assistant Editors
Gerry Heine (214)837-7268
Archie Pinkney (214)943-7710
Alan Lintol (214)220-8285
Alan Kirby (214)381-2707

Newsletter Exchange Editor
Tom Prickett (214)690-9087

Advertising Director
— o p e n —

The opinions expressed herein are those of the authors and do not necessarily reflect those of the Group or its members. Copyright © 1989 by North Texas PC NEWS. (Articles without specific copyright notices may be reproduced unchanged by other User Groups if credit is given to the author and the publication.)

Deadline:

All advertising and other material for publication in North Texas PC NEWS must be received by the NEWS staff by the 10th of the month prior to publication. See copy deadline below.

Circulation:

North Texas PC NEWS circulation is 1550.
Member distribution is 1264; remaining copies are distributed to PC user groups around the country, and to advertisers, prospective members and others with common interests.

This issue of North Texas PC NEWS was composed using Xerox Ventura Publisher. Coral HEADLINE was used for some headings. Repro was printed on a NEC LC-890 Laser Printer. Typefaces include: Palatino, Helvetica and Zapf Chancery.

DEADLINE

**Copy deadline for November
NT PC NEWS:
Tuesday, October 10th**

Meeting Dates:

October Meeting - 2nd Sat (14th)
November Meeting - 2nd Sat (11th)
December Meeting - 3rd Sat (16th)
(tentative)

*Thanks for the articles —
keep them coming!*

Submitting Articles for Publication in North Texas PC NEWS

1. Article Style. Type all copy flush left without justification. This includes headings, bylines, and the first line of each paragraph. Place a credit byline (author's name) between the title and first paragraph. Leave a blank line between paragraphs.

2. File Formats. ASCII text files are preferred. Use .TXT extension for ASCII files. If formatting is crucial, Microsoft WORD and WordPerfect files will be accepted. Other word processor file formats may be acceptable, if the article is accompanied by hardcopy and an ASCII file version of the article. Word processor files create a lot of extraneous work for the editors. If the article can be ASCII-fied, please do so.

3. Submitting Articles. You may use one of three methods.

a) NTPCUG BBS (Preferred). Log-on to the BBS and select (U)pload from the main menu. Your default file transfer protocol will be displayed. If you want to change your default protocol, use the (P)rofile option. Once you have set the file transfer protocol, select the (A)rticle option from the upload menu. You will be prompted for the filename to upload and a one-line description of

the file. Enter the filename and the description and begin the file transfer. (OPTIONAL -Send a BBS mail message to Douglas McQuaid regarding your article.)

b) Hewlett-Packard BBS. (214-830-6360 N-8-1). If you get garbage when you dial up, transmit a break (Alt-F7 in ProComm, Alt-B in ProComm Plus). login: should appear. Type ntpcug (all lower-case). Immediately you will see password: Type news (lower-case). You will get a welcome message and the NTPCUG> prompt will appear. All commands must be lower-case only. If you need help, type hints. The H-P BBS supports XMODEM and KERMIT protocols. For details type xhelp for XMODEM help or type khelp for KERMIT help. Examples of use are in each help file. Upload the file. Then type submit followed by the filename and <ENTER>. To log-off type <Ctrl-d> and give your modem a command to hang up. Please be sure to log-off (by typing <Ctrl-d>) or you will lock-up the modem at H-P.

c) SneakerNet. Track down one of the editors at the monthly meeting and give them a diskette with the article on it.

Table of Contents

President's Message	1
Jim Hoisington	
Introduction to 80x8x Assembler	5
Fred Williams	
.BAT File Discovery	10
Jim Hoisington	
On Sharing... The Sequel	11
Connie Andrews	
New and Improved System for Submitting Articles	15
Douglas McQuaid	
On Complexity, No. 32	21
Jim Hoisington	
Convert your XT into a Stealth AT	23
Zack Porterfield	

Special Items:

Agenda	1
Edlines	3
SIG Reports	14
The Variety Store	16
Swap Shop	18
CCD News	20
New-user Class Schedule	20
Volunteer Guide	22
Membership Application	24
Officers, Meetings & Times	25

Program for October 14 _____ Timothy Carmichael _____

9:00 AM - 10:00 AM

Syscorp International

MicroSTEP

Speaker: Thomas Meadow, Senior VP

MicroSTEP is an application generator (case tool) which generates C source code. Programmers should take note of this new tool, which creates your C program after you define the functions with a menu-driven query system.

10:00 AM - 11:00 AM

Good Software Corporation

Arriba - The Versatile Personal Information Manager

Speaker: Elliott Lowe, Director of Marketing

Good Software is a local publicly held company which develops and publishes PC software for IBM and Macintosh markets. Their newest product, Arriba, is being rated by leading PC publications as the friendliest PIM on the market. PIMs are growing in popularity because they bridge the gap between word processors, databases, and project management products.

1:00 PM - 2:00 PM

Prodigy Services

New Dynamics in Interactive Online Services

Speaker: Jan Edwards, Sales Rep.

Come see a live demonstration (with a large-screen VGA projector system) of this new on-line service. A flat monthly rate gives you access to headline news, detailed news, sports, weather, shopping, easy Sabre, stock market quotes, educational games, and many other services.

Prez Sez...

Free Software

I bet that got your attention. Prodigy, a new telecommunications service, will be with us in October. We hope to have one of the rooms on the first floor near the auditorium set aside for them. They will be giving away the software to access the service to all our members.

Not so free Software

At the April meeting, four of our members "won" copies of Microsoft Excel. The people from Microsoft took our member's names and addresses and promised them that the software would be shipped as soon as they got back to Redmond. I'm writing this in September and they still haven't sent the software!

It's not that I didn't try. I've made four long distance telephone calls and written a letter but our members still do not have their software. The official word from Microsoft is that Excel is currently "out of stock." Right! And the check's in the mail!

Membership Database

We reached another milestone in early August; we passed the 1,300 member mark.

I took over the job of membership director in July. John Mackoy, our previous membership director, had been working with the membership database using the copy of Paradox 3.0 that Borland gave us.

There were still quite a few reports and procedures to be completed and very little time to finish them. Because I use Advanced Revelation in my consulting practice, I imported the Paradox data into AREV (Advanced Revelation). Since then I have completed most of the things that we have been trying to automate.

For example, when you receive your membership renewal letter, you will get a printout of your current database record rather than a blank renewal form. All you have to do is mark any changes and return it with your check. That's easier for you and easier for the person doing the data entry.

The renewal letter mail merge is now created directly in WordPerfect form rather than having to undergo a conversion. (Both Zack Porterfield, President-Elect and I use WordPerfect.)

Volunteers for the information booth, equipment setup, etc. are given to the respective committees in the form of small dBase files. This means that the people in the committees can use dBase, PC-File DB, Paradox or any other database that can read dBase format files to manage their volunteers. ►

IT'S HERE !!!

**The PRODIGY SERVICE is now
available in the Dallas/Fort Worth area !!**

Attend the September 23, North Texas PC Users
Group and receive your

FREE !!

PRODIGY SERVICE START-UP KIT

AND

a 90 Day Membership !!

Over 700 features:

- ★ make your air, hotel and car reservations
- ★ 15 minute delay stock quotes, buy/sell stock
- ★ communicate with members across the country
- ★ on-line shopping, check consumer reports



PRODIGY[®]
Interactive Personal Service

1-800-822-6922 ext. 588

PRODIGY is a service mark and registered trademark,
and PRODIGY Interactive Personal Service is a
product of, Prodigy Services Company.

Membership cards are created electronically for transmission to the person that has a printer capable of printing them.

We're still formalizing and documenting the procedures but we hope to have the job done by the end of this month. I am telling you this because as we grow in numbers, 1) it takes more and more work to make this organization function, and 2) we need to take advantage of all the help our computers can give us.

Finally, let me say that I could have done all this with Paradox. It is a very capable database. However, I needed to get the job done in the shortest amount of time possible so I went with the software I know best. We will continue to use Paradox for the demographics survey and other applications.

BBS Changes

One of our most labor intensive procedures in the past has been the Bulletin Board registration procedure. Beginning in September, we have made the process a lot easier and faster.

To gain access to the BBS, you must now register on the BBS itself. Just call the BBS and follow the procedures for new registrations. If you've been a member for at least a month, your registration should be validated within a week. If you just joined, it may take a little longer because we have to generate your membership number. Your initial password will be your membership number. We're still working on the procedures to pass this information from the membership database to the group that administers the BBS, but when we finish, by the time you receive your membership card, you will be eligible for BBS access.

Jim Hoisington ▲

EDlines



This issue's deadline was almost two weeks prior to the September meeting. Despite that fact, many writers came through for us. So, here's what you will find in this issue.

In his column, On Complexity, Jim Hoisington tackles the subject of error messages. If you think Abort, Retry, or Fail is scary, you should read about the way things used to be.

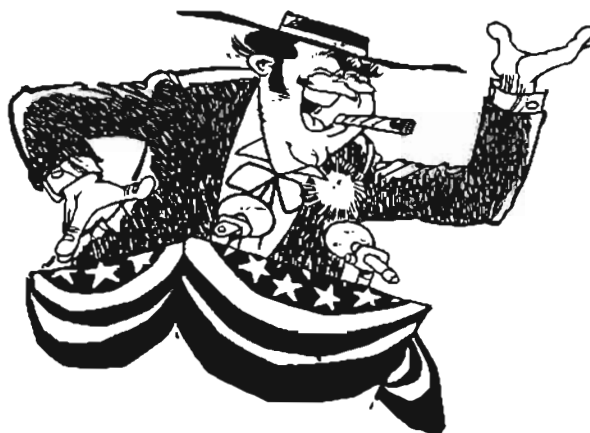
Zack Porterfield discusses converting an XT into an AT class machine. A timely subject which will interest many of you considering such an upgrade.

Fred Williams contributes an introduction to 80x8x assembler coding. The example code is a handy batch file command, ASK.COM, that polls the keyboard for input. Fred's wit shines through and this makes for fun reading even if you don't know assembler. The DOM group said that they will have a DOM disk (#413) ready to sell at the meeting. This disk will include the source code and a ready-to-run ASK.COM file.

For those of you who would like to contribute to the North Texas PC News, I've written a description of the new procedure for submitting articles via the NTPCUG BBS. Feel free to write a how-to article or a hardware or software review and upload it. We welcome input on computer related topics. Please contribute.

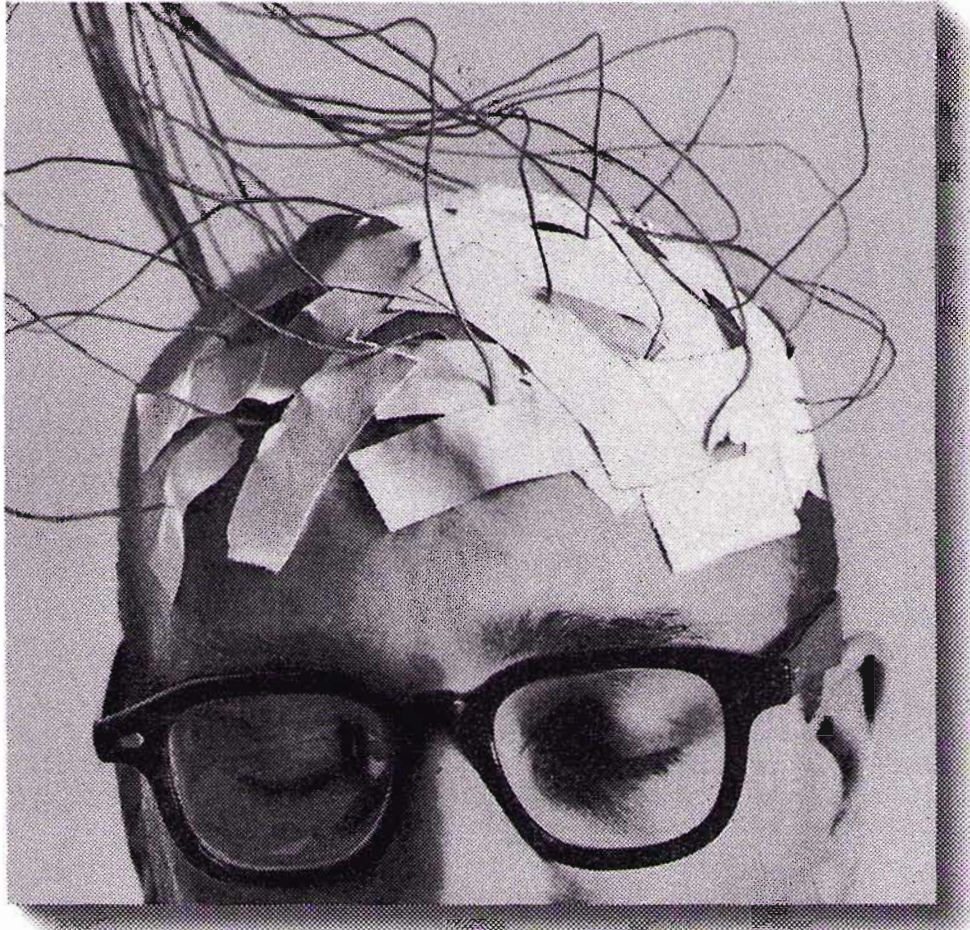
Also, we have SIG Happenings, Prez Sez, CCD Access, and other regular features. See you next month.

Douglas McQuaid ▲



*Election of officers
is coming soon...*

Until now there was only one way to integrate C and Assembler.



While C and Assembler give you power to burn, switching back and forth between them can leave your brain feeling a little fried.

All that stopping. And starting. And constantly retracing your steps.

Well, relax. Now there's Microsoft® QuickAssembler. Available with our clever QuickC® Compiler in one location: the first integrated environment for C and Assembler.

For the first time, you can save time with an integrated editor, compiler, assembler and debugger that let you create C programs, mixed C and Assembler programs, or Assembler programs that stand alone.

To make sure you feel at home in your new environment, we've designed Microsoft Quick Advisor, a hypertext electronic manual that coaches, coaxes and guides you on screen.

Quick Advisor gives you access to information on all ROM BIOS and MS-DOS® calls. And it even lets you cut and paste sample programs, so you can make both C and Assembler subroutines part of your routine in no time.

For more details on the incredible integrated power of QuickAssembler and QuickC Compiler, call (800) 426-9400. If you own QuickC Compiler version 2.0 already, we'll tell you how to add on QuickAssembler quick.

And take a load off your mind.



Microsoft®
Making it all make sense

Introduction to 80x8x Assembler

Fred Williams

I been going to do this for quite some time now, but have never found the proper "vehicle" on which to base the article. I have been looking for a simple, small, straight forward, and most of all, useful piece of code to use as the basis for a good introduction to assembler programming.

I was in one of my current client's office the other day, and had an occasion to look at one of the batch files on the network. Well, in this particular batch was a command, "ASK". After examining the operation and function of ASK, I decided it would be nice to have a copy of the thing for my own use. That's when I found out that it was supplied by our man, Peter Norton, as one of his DOS utilities. Drat, being a high minded software developer who stands far above software piracy, (even on a trivial scale), I allowed I'd just write my own "ASK". Hopefully Mr. Norton won't hit me with a "Look and Feel" suit. Boy, the land of software is for sure a legal quagmire! Sad part is, there seems to be no fair and easy solutions.

Well, enough lawyer gruel. Anyway, Norton's ASK is an .EXE file, probably so it will run on OS/2. Yech! It also takes about 1.2+K of disk storage (must be written in "C"). Well, we can do better than that! Our ASK ends up as a .COM format file requiring only 591 bytes of storage. Yes, Virginia, you could write it as an .EXE file, but not today. Maybe in another article.

On with the task at hand! First let's talk a bit about how ASK works. I've, on many occasions, wished that the DOS batch language(?) had a command that provided at least some simple level of user interaction which would allow interactive control of batch file operation based on the user's current needs. The PAUSE command's "Press any key to continue..." is all you get in the standard DOS BATCH. The PAUSE command is very helpful in determining if the user is still breathing I guess, but not much else.

ASK allows you to "prompt" the user with a message, and then wait for his selective response. ASK returns a DOS return code (referred to as ERRORLEVEL in BATCH command syntax) whose value is based on the key pressed by the user. Should the user press a key which is not a member of the valid key list, ASK "beeps" and waits for another key press. If the user presses the [Esc] key, ASK will return a return code value which is one greater than the highest valid key code. By examining the DOS ERRORLEVEL immediately after the ASK command in a BATCH file, you may selectively control the next sequential event to

occur. It ain't great, but it beats the socks off of PAUSE!

I have included an example DOS batch file, "TSTASK.BAT", for demonstrating the function of ASK and to provide an example of the command usage. If you mess up ASK's command line syntax, ASK will tell you at execution time, so you might want to test a newly created batch file using ASK before shipping it half way 'round the world. Basically, the ASK command line consists of the "prompt" string, enclosed in quote marks (") immediately followed by a series of up to 50 valid keys, with no intermingled spaces or separators.

For example:

```
ASK "Please select [A|B|C|D|Z]: "ABCDZ
```

When the above ASK command is executed in a batch file, the user will see:

```
Please select [A|B|C|D|Z]:
```

with the cursor positioned one space to the right of the colon.

The user may respond by pressing any one of the listed keys: "A", "B", "C", "D", or "Z", or the "Esc" key. Should the user press any other key, the system will "beep" and wait for another key press. When a valid key has been pressed, ASK will terminate and return a return code whose value is directly related to the pressed key's position in the key list in the invoking command line.

For example, if the user presses the "A" key, ASK will return a return code value of one (1). Should the user respond with a "D", ASK will return a value of four (4) in the DOS return code; and if the user presses the "Esc" key, ASK will return a value of six (6), which is one greater than the last valid key code's value, that being the key "Z". You will notice that "Z" occupies position number five in the valid key list, counting from left to right.

Like I said before, this little function is small and useful, and in addition, we will depend heavily on most of the 80x8x processor's "string" operands to help us get the job done with the least amount of code. For an excellent explanation and code examples of the string opcodes and for a just plain good introduction to 80x8x assembler coding I highly recommend Microsoft's "Macro Assembler 5.1 Programmer's Guide". If you are new to assembler or do a lot of debug time, you should take the time to read the book, as it seems to be well written.

Also, Microsoft's authors make a big deal out of not directly supporting the ".COM" file format. So what? I'll show you all it takes to make a small, well written ".EXE" format file into a ".COM" file using DOS's dumb old DEBUG command. By the way, this program WILL NOT work right if you fail to convert it

to the ".COM" format. Hopefully, when we are done, I won't have to tell you why.

During my annotation of the code please refer to the printed source listing presented elsewhere in this article. I'll cover each section of code in detail as to function. I will also cover any of the less than straight forward working instructions' operation in detail, but I don't intend to teach you how a move, jump, or other simple instruction works. That SQL series 'bout did me in! If you don't possess at least the very basics of the language, please read the Microsoft books supplied with Macro Assembler 5.1 first.

What you will learn from this article is some of the details of how DOS works and some of the power contained in the 80x8x string instructions. Also, I will discuss alternative coding methods and what might be gained or lost by implementing those variations.

One other reference I might recommend is the Waite Group's, "MS-DOS Papers". This is a pretty good reference guide to some of the DOS features which are of greatest interest to programmers. I got my copy at the Bookstop at Forest and Preston Road in North Dallas. I think the book publishers have figured out that books are like dope to computer techies. Have you checked the current price of the K&R C Language Reference? Unbelievable! I'm still resisting buying darn near a hundred dollars worth of CICS Cobol books (2). Talk about gutter level reading!

Since ASK gets its control information from the DOS command line, I'll give a thumbnail sketch of how the DOS command line works. When COMMAND.COM services a command line program run request, it parses any parameters entered on the command line following the program name into an area of the requested program's memory load space known as the Program Segment Prefix, or PSP. The PSP is contained in the first 100h, (that's 256 for you decimal types) bytes of the load module. Please note the "ORG 100h" statement immediately following the assembler assume directive. The effect of the ORG statement is to insure our program code does not overlay the PSP or vice versa.

Another note is in order about the program's memory model. All ".COM" files must be coded as a "Tiny" memory model. This means that all the code, data, and stack must fit into a single 64k memory segment (I think 591 bytes might fit). As I said earlier, Microsoft does not directly support the Tiny memory model with one of their fancy .MODEL statements. They do at least give you a fair example of a small Tiny model program in the Programmer's Reference. This means that we will code all of our data buffers and code in a single program segment, which we have named " _code". We also code the "ASSUME" register directive to indicate that all of the

segment registers will reference this single segment. Once the initial housekeeping is done, we jump to the actual first instruction, "JMP START", which is located past our program's own data buffers.

In our data area ("WORKING STORAGE" for you poor Cobol devils) we have set aside 50 bytes of storage for our incoming valid key list. One word, "TBLEN", will serve the dual purpose of providing the basis for the [Esc] key's return value, plus provide a length value for the actual length of our valid key table. Next we have set aside an 80 byte long buffer, which is where we will save our incoming "Prompt" string. Then we have storage related to the "Usage" message we will send when someone fouls up the command line syntax. And last is the ASCII "Bel" code used to beep the speaker. You might notice that all of the displayable character strings terminate with the "\$" character. This is the "stopper" character required by the DOS display service interrupt we will use to send strings to the standard output device.

One thing you may notice while wandering through this code is that my old "structured programming" habits tend to follow me around and haunt me. The first real instruction in the program is a "CALL" to a procedure which will parse in the content of the command line for us. Why didn't I put that as inline code? Well, what can I say? If you wish, move the parsing routine inline and save another few bytes of storage by trashing the "CALL" and related "RET" statements. Sloppy, sloppy, sloppy! And I'm suppose to be showing you.

The things we need to do to parse in the command line is: (1) Establish what to look for (the first occurrence of a quote (") character). (2) Establish addressability to the command line in memory (located at position 80h (128 decimal) in the PSP. (3) And, establish a maximum length to search (80h by DOS definition).

Now to efficiently scan the table, we are going to use one of the 80x8x's string opcodes, "SCASB". The SCASB instruction searches a string in memory who's first byte is located at the location pointed to by the register pair, ES:DI, for a length of the content of CX. By using the Opcode prefix of "REPNE", SCASB will set the processor's "Zero" flag if a character matching the content of register AL is found in the string being searched. If no match is found, the processor's Zero flag will not be set when the search terminates at the end of the string.

If a matching character is found, the position of the first occurrence of that character in the searched string, plus one, will be returned in register DI. So, if there is a valid command line, we want to save the contents of the DI register, as it points to the first character in our "Prompt" string in the command line. If no beginning quote character is found, the

►

command line is bad, and we jump to the routine which will send the usage message to the standard output device and terminate ASK.

Now that we have the first character of our prompt string located, we use another REPNE SCASB instruction to locate the ending prompt string quote character. If no closing quote is found, jump to the usage message display and terminate routine. If an ending quote character is found, move the closing quote character's location, plus one, to the CX register. Recover the address of the first character in the prompt string by POPping it off of the stack into the DI register. We then save this location pointer to the command line, (PUSH CX), as we will need the current address later to continue our scan for valid keys.

Next, we compute the length of the prompt string by subtracting the starting address from the ending address, plus one. Then decrement CX by one to take care of the plus one problem. Now that we have computed the length of the upcoming move, we can finish our preparations for moving the prompt string from the PSP command line to our program's prompt message buffer, "MSG", by establishing the proper source and destination addressability.

To perform the prompt message move, we first load register SI with the starting address of the message buffer area, MSG. We then XCHG (exchange) the SI and DI registers so that the register pair ES:SI points to the source string (the command line prompt string) and DS:DI register pair points to the destination (first byte of our program's message buffer, MSG). Register CX already holds the computed length of the upcoming move.

The actual move is accomplished using the 80x86 MOVSB instruction prefixed with a REP. This moves a number of bytes equal to the value stored in register CX from the source string to the destination string. The move terminates with register CX equal to zero, and both SI and DI pointing to the byte which is one byte location past the last byte of the source and destination strings, respectively.

To properly terminate the prompt string for subsequent display on the standard output device, we move a dollar sign (\$) to the byte location pointed to by the DI register. Remember that this is the byte location that is one byte location past the end of our just moved prompt string. We do this by loading the dollar sign character into the AL register and then execute the STOSB instruction which stores the byte in register AL in the memory location pointed to by the ES:DI register pair.

With our prompt string now safely in our program's buffer area, we may proceed with scanning the remainder of the command line for our list of valid key codes. To restart the parsing, we first load the

address of our valid key code table into register DI. We then POP, into register SI, the location of the byte immediately following the closing quote mark of the prompt string in the command line. Remember, we pushed that address pointer on the stack prior to moving the prompt string. We will need to count the bytes that are stored in our valid key buffer, so we initialize register DX with a zero by doing an "XOR DX,DX". Now we are ready to parse in the valid key codes.

To move the valid key codes, we will use a pair of special 80x86 load and store string Opcodes, LODSB and STOSB. Execution of a LODSB causes the byte addressed by register pair DS:SI to be loaded into register AL. LODSB also increments SI by one. STOSB execution causes the byte stored in register AL to be stored in the memory location pointed to by the ES:DI register pair. STOSB also increments the DI register by one. Now that we understand the instructions we are going to use, let's step through the valid key code string parsing loop one time.

In the key code parsing loop, we first load the currently addressed key code located in the PSP command line into register AL. To do this we use the LODSB instruction. Remember that LODSB increments the source string pointer, register SI, by one. Next we compare the contents of register AL with the hexadecimal value 0D. (You all know that is the hex value for an ASCII Carriage Return character, right?) Why? Well the command line in the PSP is terminated with an ASCII Carriage Return character, that's why. So, when we find a hex 0D in the AL register, we will know there are no more valid key codes to move.

Oh no! Immediately after our check for the ASCII Carriage Return character, Mister Structure strikes again! There is another CALL statement to a routine which will convert any ASCII lower case letters to upper case. Well, gotcha! This one is quasi legal, as this is a routine that will be used elsewhere. So, this way we write it once to save on the code byte count, and CALL it when we need it. Yes, you performance freaks could put it inline and save the time used by the CALL and RET instructions, but I really doubt the user will notice the increase in performance. Plus it will take COMMAND.COM longer to load your bigger program. So, pay me now or pay me later.

Once we've converted lower case to upper case, we use the STOSB instruction to store the character in register AL in the memory location pointed to by the ES:DI register pair. And, don't forget that STOSB increments DI by one also. Next, we increment our key code counter by one and jump back to get the next key code. This loop will continue until we find a hex 0D in the AL register, telling us that we have reached the end of the command line, and we have processed all of the valid key codes. ►

The last thing to be done in the command line parsing routine is to save the key code count we have accumulated in register DX. This is the length of our key code "string" which we will search each time the user presses a key. So with our key code string length saved, we can return to the main line of our code and get down to the business of checking key presses.

After all the work we went to initializing the program, the actual main line code is going to appear kinda' simple. First thing we do is establish addressability to the prompt message, and CALL a routine to display it on the standard output device. Once we have displayed the prompt string, we will enter a loop that will receive and check key codes until the user presses a valid key or the [Esc] key.

First thing we do in the mainline loop is tell DOS to clear the keyboard buffer and wait for a key press. This is done using the DOS hex 0C service request. This DOS function request clears the keyboard buffer and immediately executes an additional DOS function request whose value is passed in the AL register. In our case, we have placed 08 in register AL, which tells DOS to wait for a key press and return the value of the key pressed in register AL immediately without "echoing" the key code to the standard output device.

For those of you who would like to have the pressed key appear on the standard output device, you may place an 01 in register AL instead of the 08 I used. DOS function call 01 works exactly like function 08, except the key pressed is displayed on the standard output device. Both of the functions, 01 and 08, recognize the special [Ctrl Break] key sequence.

When DOS has returned the key code, we check to see if our user has pressed the [Esc] key, "CMP AL,1BH" ([Esc]'s key code is a hex 1B). If the key pressed was not the [Esc] key, then we continue. If it was the [Esc] key, then we increment the key code string length value by one. This makes the value one greater than the highest legal key code value. We then move this new value to the AX register and jump to our program exit routine.

If the current key pressed was not the [Esc] key, we CALL the lower to upper case conversion routine. Then we load register DI with the address of the beginning of our valid key code buffer, load register CX with the valid key code string length, and execute another SCASB with a REPNE prefix. Remember how this worked in the parsing routine? If SCASB matches the pressed key code saved in register AL with a key code in our valid key code table, it will end execution and set the processor's Zero flag. If no match is found, the Zero flag will not be set.

If no matching key code is found, we jump past our valid key code logic, load register DX with the ad-

dress of our speaker beeping string, CALL the standard output device display routine, "PMSG", and JMP back to wait for the next key stroke. If a matching key code is found, our user has pressed a valid key, and we are ready to establish a return code value for the pressed key and terminate.

To establish the return code value for the pressed key, we need to do a bit of math. First we load register SI with the starting memory address of our valid key code table. Then we subtract the value contained in register SI from the contents of register DI. This leaves the position value of the pressed key code in register DI. Remember that the SCASB instruction has left the contents of register DI with a value which is one greater than the location of the matching key code in our key code table. So, when we subtract the beginning buffer address from the value of DI, we will end up with a value that is equal to that key code's position in the key code table.

We then move the value in register DI to register AX and jump to our program exit routine, "BYE". In the program exit code, all we need to do is load register AH with a hex 4C, the terminate and return the return code value loaded in AL, DOS service request. We then issue the DOS service request interrupt, interrupt 21H. This will cause ASK to terminate, free its allocated memory, and leave the DOS return code set to the value of the pressed key code's table position.

Well, that wraps up the code walk through. I didn't cover some of the simpler things. But, that will give you something to study and learn about. I think one of the personality traits that a good programmer must have is a strong desire to study and understand the detailed workings of most anything he runs across. It is the details that make or break program code.

Now that the program is coded, we need to Assemble and Link it. I'm not going to teach you how to run the Microsoft MASM assembler or the DOS Linker. I said I'd introduce you to Assembler, not take you to raise! Once you get all of the stupid coding errors out of your source code, it will assemble clean. Next, you should link it using the DOS program linker LINK. LINK will issue a warning "LINK : warning: no stack segment". Ignore it, we already know that. And don't care, as we are creating a Tiny model program which has no separate stack segment.

Don't execute the resulting ASK.EXE file. It will only hurt your feelings when it crashes and burns due to addressability problems. The next step is to dust off the old DOS DEBUG. Load ASK.EXE into DEBUG:

```
DEBUG ASK.EXE
```

Change the file name to ASK.COM with the DEBUG name command (n). Like so: ▶

-n ask.com

Then tell debug to write out the code, with the DEBUG write command (w):

-w

Then exit DEBUG with the Quit command, (q):

-q

Now, just to be safe, delete the ASK.EXE file. We don't need it anymore, and don't want it around. DOS could end up executing it by mistake in the right set of circumstances, which could prove to be, at least, most perplexing.

What is now left is our ASK.COM program ready to be tested. That's where the "TSTASK.BAT" file printed somewhere in this article, comes into play. I'm not going to cover the workings of TSTASK.BAT, other than to explain how the DOS Batch IF command checks the DOS return code. Good Grief, if you don't understand DOS Batch, how do you expect to write assembler code? The reason I'm going to explain the IF command is because I shot myself in the foot with it, and thought you might, if you aren't careful.

The way the DOS Batch IF command checks the return code is kinda hokey, but executionally cheap. The IF command compares the DOS return code (ERRORLEVEL) with the value specified in the conditional phrase of the command and branches if the

value of the DOS return code is equal to or LESS THAN the value specified in the IF conditional. So, notice how I laid out the IF commands in TSTASK.BAT. And, that's not how I did it the first time.

Now a little 20/20 hindsight. I mentioned some performance and code saving issues during the code walk through. But I'm wondering if anyone has spotted one of the most obvious ways to make dramatic improvements in program size? Well, I'd bet a savings of right at 130 bytes for sure. I will mention that, once your program is in control, the command line part of the PSP is no longer sacred ground.

That all folks! The DOM group will have a DOM Disk available which contains the MASM assembler source for ASK, the TSTASK.BAT file, and a working version of ASK.COM available for purchase, as I'm releasing this work of programming art as a freebie. I included the ASK.COM file for those of you who are too lazy or too smart? to mess with learning assembler, but wish to have the ASK command for your DOS batch files. Oh yeah, and for the poor slobs who never manage to get a clean assembly, before they give up their budding career as a scribe of GOD's Language. Oh well, I've heard there is still a big demand for CICS Cobol hacks.

Fred a

(Fred Williams is the owner of Systems Consultants, a data communications software development, networking design, and consulting firm.)

ASK.ASM

(Source Listing for ASK.COM)

```

title ask 'DOS Batch utility'

; Batch file utility which pauses the batch file operation and
; allows the user to keyin a key stroke response. ASK then exits
; with an ERRORLEVEL value equal to the position of the key code
; in the command line used to invoke ASK. The [Esc] key will
; always return an ERRORLEVEL value one higher than the highest
; valid key code. Invalid key strokes will cause a "beep".

(C) Copyright 1989 Systems Consultants
ALL RIGHTS RESERVED

Released, free of charge, for personal use of recipient only.
Not for resale or redistribution, either singly, or incorporated
in other products, for financial gain in any manner.

WARNING:
Must use DEBUG to rewrite the file as a ".COM" format, for proper
function.

_code segment
assume cs:_code, ds:_code, es:_code, ss:_code
org 100h

begin:
jmp start ;must be a FORDI

;
; valid key code table
; entries are stored in
; command line sequence
; [Esc] key value-1 & tbl length
tblen dw 0
msg db 80 dup (00) ;prompt message buffer
umsg db 13,10 ;start of "Usage" message
    
```

```

db 'Usage: ASK "<message>"<key1><key2>...<key50>'
db 13,10,'$' ;end of "Usage" message
beepit db 07,'$' ;invalid key beeper

;
; start:
call scan ;scan and store command line
lea dx,msg ;point to the "prompt"
call pmsg ;go print it

getkey:
mov ax,0c08h ;clear the kbd buffer and wait
int 21h ;for a key
cmp al,01bh ;is it [Esc]?
jne notesc ;not the [Esc] key
inc tblen ;make one higher than last key
mov ax,tblen ;get the [Esc] return value
jmp short bye ;and split

notesc:
call uplow ;convert lower to upper
lea di,seltbl ;point to the valid key codes
mov cx,tblen ;set the table length
repne scaasb ;look for the key code
jnz notfnd ;invalid key code
lea si,seltbl ;point back to the start
sub di,si ;subtract start from current
mov ax,di ;result is return code
jmp short bye ;out 'ah here!

notfnd:
lea dx,beepit ;wrong key beeper
call pmsg
jmp getkey ;go try again

;
; usage:
lea dx,umsg ;screwed up the command line
call pmsg ;message

bye:
mov ah,4ch ;exit with return code in AL
int 21h ;back to DOS
    
```



```

;
pmsg:  mov ax,0900h ;display a string on
      int 21h      ;the standard output device
      ret

scan:  mov al,22h    ;ASCII quote (") character
      mov di,80h   ;start of command line in PSP
      mov cx,80h   ;max length of command line
      repne scasb  ;look for a quote
      jnz usage    ;no prompt message found
      push di      ;save the first char pointer
      repne scasb ;look for the ending quote
      jnz usage    ;missing ending quote
      mov cx,di    ;current location
      pop di       ;recover first char pointer
      push cx      ;save current location
      sub cx,di    ;length = (end +1) - start
      dec cx       ;adjust to true length
      lea si,msg   ;point to the "prompt" buffer
      xchg si,di   ;flip for the MOVSB
      rep movsb   ;and move the string
      mov al,$    ;terminate with a "$" for DOS
      stosb
      lea di,seltbl ;point to the "valid codes" table
      pop si       ;recover command line pointer
      xor dx,dx    ;init a counter to zero

more:  lodsb        ;get a key code from command line
      cmp al,0dh   ;end of command line?
      je nomore   ;yes
      call uplow  ;convert lower to upper,
      stosb       ;store it in the valid codes,
      inc dx      ;and count it
      jmp more

nomore: mov tbllen,dx ;save the actual table length
       ret

uplow:  cmp al,61h   ;convert all lower to upper
       jb notlower
       cmp al,7ah   ;
       ja notlower
       and al,0dfh  ;lower to upper

notlower: ret

_code  ends
      end begin

```

TSTASK.BAT

(Batch file for Testing ASK.COM)

```

echo off
rem
rem Batch file for ASK command function
rem check out.
rem
:menu
ask "Please select (A)|B|Z|Esc): "abz
if errorlevel 4 goto excp
if errorlevel 3 goto az
if errorlevel 2 goto ab
echo You pressed the (A) key.
goto menu
:ab
echo You pressed the (B) key.
goto menu
:az
echo You pressed the (Z) key.
goto menu
:excp
echo You pressed the [Esc] key.

```

.BAT file discovery

Jim Hoisington

I discovered something about BAT files this week that I had never known. I discovered that you can use Environment variables in BAT file IF statements.

The Environment is a block of memory allocated for each program in memory. Even COMMAND.COM has an Environment block allocated to it. The default size of the Environment block varies with each version of DOS but if you want to use it, the default size is almost always too small. The size of the Environment block can be increased by using the /e: option on the SHELL command that goes in the CONFIG.SYS. Like the default size of the block, the method of increasing the size using the /e: option varies with each version of DOS, so you need to consult the DOS manual for the version of DOS that you are using.

The reason that I want to use environment variables in .BAT files is that I currently have a wide mixture of computers on a local area network. Some have graphics screens and some don't. Some have a mouse and others have a math coprocessor. If I run a program in the AUTOEXEC.BAT file that creates the environment variables: VIDEO, MOUSE and FPU, I can then keep people from trying to run programs that require a graphics card, mouse or math chip (Floating Point Unit) on computers that don't have that equipment.

For example, consider the following .BAT file for starting EasyCad:

```

IF %VIDEO%==MDA GOTO :NOGRAPH
CD ECAD
ECAD2
GOTO :EXIT
:NOGRAPH
ECHO You must have a graphics card to run EasyCad
:EXIT

```

The program run from AUTOEXEC.BAT creates an Environment variable named VIDEO and sets it equal to MDA, HERC, CGA, EGA, MCGA, PGA or VGA. To use an Environment variable in a .BAT file, you need to precede and follow the name with a percent sign. The first line of the .BAT file tests %VIDEO%. I'm sure there are other uses for environment variables, but this application will make life easier for my network users.

Jim



On Sharing, ...

The Sequel

Connie Andrews

When I acquired my first car some years back, I was a junior in high school. It was an old clunker with many, many miles on it. Not much to look at and barely ran at times, but it only cost \$200. And it was my very own. Even had a radio that would play if you hit the dash at the right spot. Bells and whistles, right?

The thrill of a first car is something special. So is the thrill of a first computer. And I have one. (Actually, I'm on my second one, but more about that later.) My first computer, acquired in the fall of 1988, was an early PC, upgraded a good deal over the years and a real workhorse. But no clunker.

Even though we are now a two-computer family, the issue of a shared home computer versus having your own still intrigues me and I would still like to come up with some words of wisdom for those who remain a one-computer family. In the September, 1987 publication of the North Texas PC News, I explored a little about what it feels like to share a computer with family member(s) in "On Sharing".

So many variables come into play when sharing a computer at home. It's a lot different than at the office, and so are the rules - spoken and unspoken.

It seems to boil down to two major areas of conflict. One is a given constant and can't be changed. That is time. Access at home is usually limited to time after work or school and on the weekend. That's a big problem when you consider all the other things you and your family must accomplish or want to accomplish in those same hours.

The second biggest stumbling block seems to occur with the different needs of beginners and advanced users. Many beginners proceed

cautiously and have some fear of the computer until they start understanding what they and the computer are doing.

To the advanced user, the computer becomes personal real quick. "Let's see here. If I set up a batch file to do thus and so, then I won't have to do it each time I turn the computer on. It's so easy and makes life so wonderful, surely the rest of the family can figure it out." You've just changed the rules. What is intuitive to you usually isn't to the beginner.

Some of our members have mentioned that they wish their family would use the computer more. They've even set up programs they think will make it easy for the family to get hands on experience. Sometimes it works and sometimes it doesn't. The principle is no different than teaching a member of your family how to play, say, tennis. A couple we know tried that and I've never seen more bitter fights. Many times teaching is better left to someone more objective and many times the person learning just needs to move at their own pace without pressure.

Making a decision about whether to invest in a second computer is difficult. If you have at least two users in the family who need or want a great deal of time on the computer, as in our case, you'd probably be wise to go for it. Born out of necessity, it's like being a two-car family. And it beats the hell out of paying for a divorce.

It's a little more difficult to justify a second computer if you only have one apparent user in the family. A second computer might encourage more participation from the rest of the family if all they need is more access time and you've been hogging it. But having a second computer will not insure more use if others in the family don't really want to use it. In that case you would end up with a very expensive door stop.

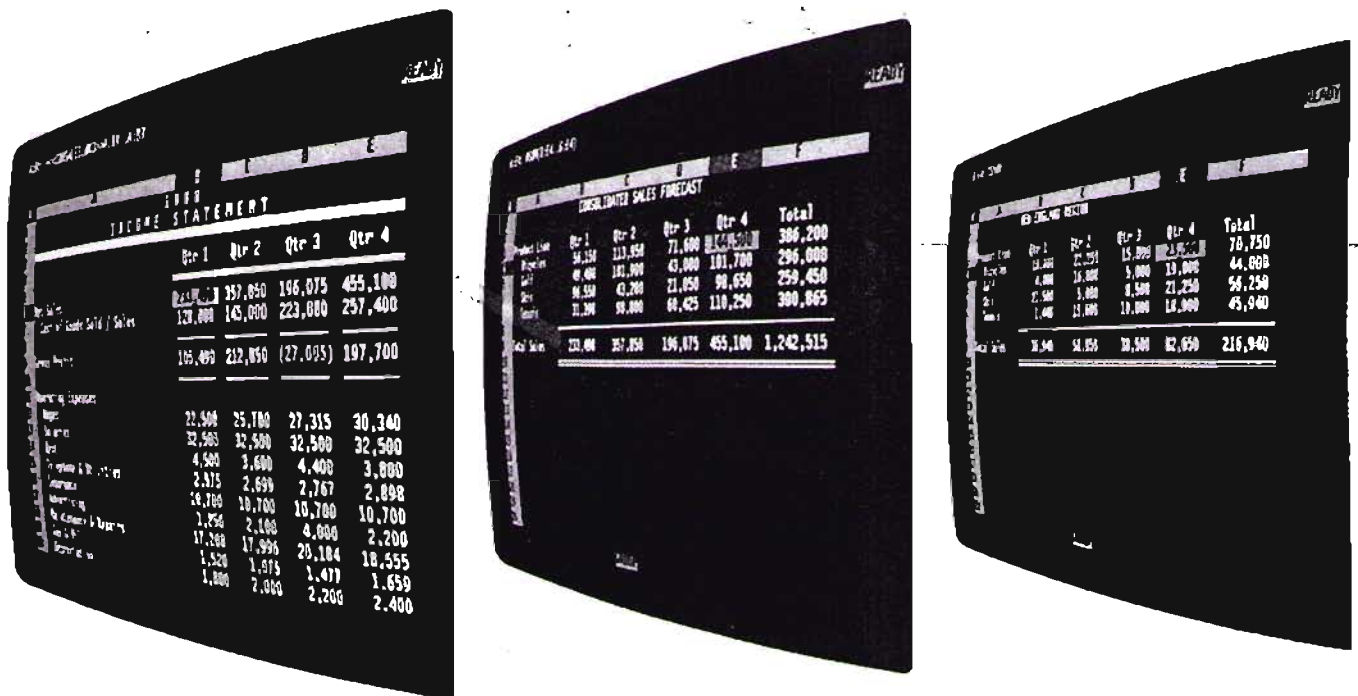
In my own case, I've spent more time this past year on the computer than I did in all the earlier years on our shared computer, even subtracting the downtime it took to troubleshoot damage from a lightning hit during one of our nasty May thunderstorms. Ended up having to replace most of the computer.

So, back to my original article. My "inviolable box" in September, 1987 has now grown from a box of floppies to a computer. Never, never land was never so good.

Connie

▲

Today, the power a lot to do with the



Release 3's 3D design opens up all kinds of new possibilities to improve the efficiency of your work.

In business, more than ever before, power is a function of information. And nothing can help you exploit the power of data like the new Lotus® 1-2-3®: Release 3 and Release 2.2.

1-2-3 Release 3 lets you manipulate data, navigate through complex applications, and work with more speed, power and ease than any other spreadsheet.

In fact, *InfoWorld* recently was moved to say "...with this new release, 1-2-3 has instantly regained its long-standing position as the sheet to beat."¹

Much of the credit belongs to Release 3's

revolutionary, true three dimensional design, which lets you view and work with up to 256 worksheets and files simultaneously. And its sophisticated file linking capabilities are unparalleled, because you can link worksheets and files in memory, on disk or on a network.

But the advantages of 3D aren't limited to spreadsheets. It also allows you to use the new relational data management capabilities to analyze information from multiple data tables. Release 3's database is simply the most powerful available in a spreadsheet.

behind a desk has power on top of it.

You can even bring data from external databases, like dBase III[®], directly into your spreadsheet. Without ever leaving 1-2-3.

Equally impressive is Release 3's presentation quality output and its selection of advanced analytical graphics.

With the Lotus Add-In Tool Kit for Release 3, you can create a wealth of customized @ functions, macro key words and turnkey applications.

Release 3 is for DOS or OS/2[®] users. It's designed to fully exploit the 1 Mb of memory that's standard on today's 286 and 386-based PCs.

Those of you who work on all types of PCs in a 640K DOS environment, read on. Because we designed 1-2-3 Release 2.2 to maximize the power in your PC.

Release 2.2 offers remarkable speed and presentation quality output, as well as improved analytical power in the way of file

The New Lotus 1-2-3

The spreadsheet of choice

linking, minimal recalc and undo.

Of course, both Release 3 and Release 2.2

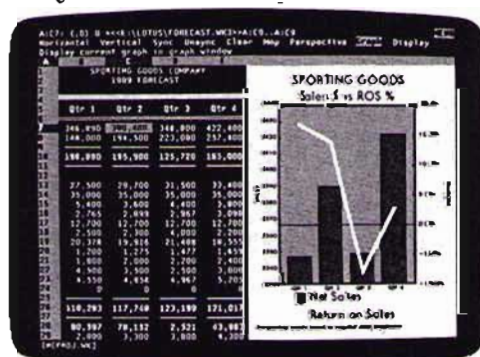
provide the highest compatibility with your present 1-2-3 files, macros and applications, and with each other.

There's never been a better time to upgrade to Lotus 1-2-3, because there's never been a better 1-2-3.

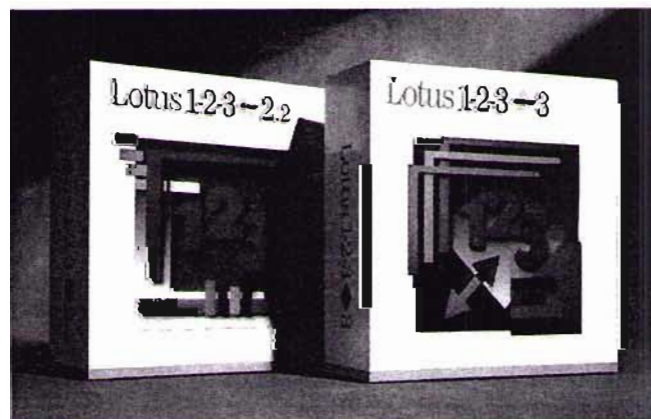
For more information, contact your Lotus Authorized Reseller. Or, if you're a current user, call 1-800-TRADE-UP.* Ask for Ext. 569 for upgrade information.

Then you'll see firsthand how the new Lotus 1-2-3 will give you more power than ever before.

On the desk and behind it.



Release 3's HotView graph window dynamically links your worksheet and related graph.



Selected SIG Happenings

News and Meeting Notes on Special Interest Groups

(Material for this column should be sent to Zack Porterfield, SIG Coordination, before the 10th of each month.)

DAC SIG

For the up and coming new DAC4 Easy Accounting Package, arrangements are being made for a Technical Support Representative to demonstrate and/or discuss the new enhancements. Everyone is welcome to come see and discuss what's happening in the World of DAC Easy Accounting. Hope to see you there.

Putt Shaw

DOS SIG

By the time you read this, IBM may have released a new DOS version, possibly version 4.1. Rumors about a new DOS are circulating around the industry with some frequency. IBM did give some endorsement to the All ChargeCard as an interim solution to PC/AT's powered by 80286 CPU's over the Summer and that may be a precursor to a new DOS version with some capabilities beyond 640K. (These are essentially the same rumors that spring up every Fall before COMDEX.)

If there is a new PC-DOS, we'll try to look at it to see what it has to offer for users. PC-DOS has a good X.1 record with DOS 1.1 and 2.1 established as "classic" DOS versions with relatively few bugs and long lives. Jim Hoisington, NTPCUG President, will explain why many users are still running 2.1, and why 3.1 didn't establish the same record with PC owners.

DOS 4.0X still hasn't caught on in large numbers, primarily because of widespread bug reports and incompatibilities with existing software. Jim will go into these issues in some depth, hopefully resolving problems in installation experienced by many

users and dissolving some of the "buggy software" myths where appropriate. October's DOS SIG Meeting will end with the usual Q & A format.

Réagan Andrews

LAN SIG

Good news! The very popular and extremely knowledgeable Bernie VanRoekel returns for the October LAN SIG meeting.

Bernie's topic will be, "LAN Design and Installation Considerations". And in his own words, "I'm putting together a collection of tips and things to watch out for - practical stuff from experience."

Bernie's last presentation was one of our best, and I expect this one to be at least as good and probably better.

Heck, he may even know a trick for assembling those engineering marvels known in more polite circles as "Token Ring Cable Plugs".

So, come one, come all, to what will surely be a most interesting and informative session.

Fred Williams

Lotus SIG

The subject for the September 1989 SIG meeting was the Range commands of 1-2-3 which include some easy command such as Copy, Erase, and Format as well as some less used commands such as Transpose, Value and Input. The presentation went well and was informative for those who attended.

The subject for the October meeting will be Data Tables or What-If tables in Symphony. Data tables are one of the most powerful features of 1-2-3 but are not often used since most people are unaware of them or are unsure of how to use them. One of the reasons they are not used is that the command is embedded in the Data menu structure which many users consider as database commands.

Data Tables add what-if capabilities to 1-2-3 without having to calculate and print multiple copies of a file. The classic illustration is a loan. A model to calculate the loan payment is very simple but requires that the principal, interest rate, and term be known. Through the use of a data table, the user could display the payment if any one of the three variables or any two variables are changed. Then on a single printout the user could have several hundred loan payments on it.

The real benefit of a data table is that the user can display the model and the methodology in one part and then shows the results if different input values are used without having to display the methodology.

The Lotus SIG always takes time to answer questions about Lotus products. If you have a question about a Lotus product or would like to learn more about Data Tables, then stop by in October.

Mark Gruner
and Pat Henley

Volunteer SIG

Last month, the first official meeting of the Volunteer SIG was held. Our objective is to help charitable and non-profit organizations make better use of their computers. At the first meeting, we discussed the basics of "How do you say 'Hello' to someone else's computer."

October promises to be even more interesting, with the topic, "What's a nice computer like you doing in a place like this?", subtitled, "1001 fun things you can do with a computer for free (or with shareware)." Offer void where prohibited by fuddy-duddies.

Jay Shilstone

WORD SIG

WORD 5.0 has some impressive graphics capabilities. Exploring and utilizing WORD's graphics features will be the topic for the October ►

WORD SIG Meeting. Problems encountered by graphics alignment and tag lines will be discussed along with work-arounds and other solutions.

New graphics features increase need for good graphics editing software. Discussion of some major graphics editors and their potential for utilization with WORD 5.0 will

be featured at the October Meeting if time permits. Ease of integration with WORD documents, ease of graphics editing and advanced capabilities are important features to WORD users and will be targeted for the discussion. Presentations on several of the advanced graphics packages are planned for coming months and SIG members input on selection of packages for review is

encouraged.

Latter portion of the SIG Meeting will be devoted to a general Q & A session focused on problems SIG members are encountering with day-to-day operation with WORD.

Reagan Andrews



A New and Improved System for Submitting Articles

Thanks to Tom Prickett, Kent Cobb, and all the folks at the NTPCUG BBS, the procedure for submitting articles to the newsletter is now almost painless. The hardest part is writing the article. Below you will see the log of a BBS session where an ASCII file (FILENAME.TXT) was uploaded to the editors. As you can see, there is now an option on the File Upload Menu to directly submit articles to the editors.

Here's a play-by-play of the new procedure:

Log-on to the BBS and select (U)pload from the main menu. At the File Upload Menu, your default file transfer protocol will be displayed. If you want to change your default protocol, use the (P)rotocol change option. Once you have set the file transfer protocol, select the (A)rticle entry option from the upload menu. From the Submit Newsletter Article screen select (A) once again to submit your article. You will be prompted for the filename to upload. Enter the filename and begin the file transfer. After the transfer, you will be prompted for a one line description of the file. Type in a suitable description and you're done! Quelle facie, n'est pas? (translation: Ain't that easy?)

What makes this new system so great is the fact that your articles will now go to an area of the BBS where the editors can get a directory listing. This was not possible before. As a security measure, the articles are archived daily and put in "safety area". We (the staff of the News) love it and hope that you will write an article and try it out.

Doug

```

-----
NTPCUG Bulletin Board
-----Main Menu-----
Enter
-----
(I) -- Information on current events
(M) -- Mail and conferencing
(G) -- General conference select
(S) -- SIG conference select
(D) -- Download files
(U) -- Upload files
(H) -- Hang up and terminate session
-----
(E) -- Expert mode select ( terse screen)
(N) -- Novice mode select (full screen)
(P) -- Profile changes (permanent)
-----
Enter letter for selection: u
    
```

```

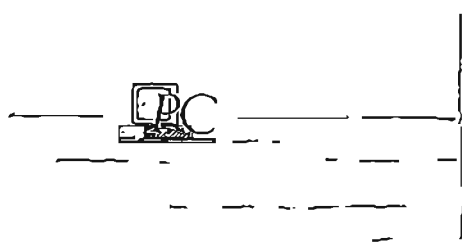
-----
NTPCUG Bulletin Board
-----File Upload Menu-----
The board provides for private exchanges of data between our users.
This is intended to provide an unrestricted exchange of data
between users. It is intended for non-proprietary data ONLY.
Data identified as proprietary will be removed.
-----
Current Protocol: KERMIT
-----Destinations Available-----
U -- (U)ser directory, files uploaded here are available
to all users who know the file name.
A -- (A)rticle entry. Upload newsletter articles
to go directly to the Editors.
M -- (M)ail, files uploaded here are available only to
users who you explicitly send the file to. You are
prompted for recipients after the upload is
completed.
-----Data Exchange Format-----
P -- (P)rotocol change.
-----
Enter letter for selection: a
    
```

```

Current Protocol: KERMIT
NTPCUG Bulletin Board
-----Submit Newsletter Article-----
Newsletter deadline is the 10th of each month.
Submit your articles with A.
Articles will go to the editors tonight.
-----
Enter (A) to submit article, or <enter> to return: a

Protocol: KERMIT
Enter upload filename (no drive/path) <enter>: FILENAME.TXT
Ready to receive FILENAME.TXT; begin transmitting using KERMIT protocol.
Press a key to continue...

Enter a 1 line description of the file:
-> This file contains an article about _____ for the newsletter.
    
```

(New or unusual hardware/software/applications for small computers.)

Lots of PC news this month. Since major "names" are active, we were afraid some of the following items might get lost in the shuffle and decided to run them first.

Out with the 3 X 5 cards!

Somewhere, lost beneath empty liners and dust jackets, is a stack of 3 X 5 cards that's yellowed with age. The cards were the start of an attempt to catalogue 300+ LP's that died before completion. *Personal Music Librarian* (PML) by Personal Database Applications may be the answer to organizing the records that index cards wasn't.

PML is a pre-defined database that accommodates definition of categories, subcategories and media composing the music collection. PML allows notes up to 64K per selection, assisted by a full-screen word processor, does complex sorting for retrieval and other operations common to good databases.

The real power here is that all the hard work is done for the user, and PML is tailored for the music lover with a large collection of records or sheet music. It's available from Personal Database Applications, 2634 Meadow Bend Court, Duluth, GA 30136-6037. Write for more information.

Rising Shareware Star

PC-File DB is scheduled for a major upgrade in November, 1989. That's not news. Buttonware makes periodic revisions to its database which you usually see in *The PC NEWS* Disk of the Month (DOM) announcements.

News here is that *PC-File DB*, Jim Button's long-lived shareware database, and its scheduled update

was handled by *INFOWORLD* just like any of the major software revision announcements. *PC-File* was one of the original shareware programs that made good software available to PC users at reasonable prices and has been a staple of NTPCUG's DOM operation since 1983. *PC-File DB* is also sold as a standard software package by Buttonware.

Late-night Idle Time Not Wasted

AutoMate is an automatic scheduler that initiates computer jobs without operator intervention and allows routine tasks to be accomplished during non-working hours. *AutoMate*'s publisher, Complimentary Solutions, Inc., describes periodic back-ups, report printing and scheduling batch jobs as examples.

Daily, weekly and monthly tasks may be scheduled by *AutoMate* which requires DOS 2.0 or higher and incorporates a 3K background TSR. A second version, *AutoMate Macro*, includes a Macro facility to assist in running programs which require keyboard input, such as communications programs, file transfers, etc. Both programs offer context sensitive help screens in three levels and pop-down menus. Price quoted is \$199 for *AutoMate*, \$249 for *AutoMate Macro*. Complimentary Software Solutions, Inc., 4470 Chamblee-Dunwoody Road, Suite 202, Atlanta, GA 30338.

Quickies...

First, from Aldus

Aldus, publisher of *PageMaker* announced a major revision of their PC desktop publishing program to allow for improved handling of larger documents. Long documents had been the province of *Ventura Publisher*, but *PageMaker* Version 4.0 will change that with release in "the first quarter of 1990"

Aldus said. Aldus also announced future OS/2 *PageMaker* versions with enhanced features, but didn't speculate on a release date.

MS WORD for UNIX in '90

Microsoft, following *WordPerfect*'s lead, announced that *WORD 5.0* for SCO Unix/Xenix would be shipping at year's end. Currently in beta, Microsoft said *WORD 5.0* would run under release 2.3 and 3.2 of SCO's UNIX OS. Reported in *INFOWORLD* in September, the Microsoft announcement was accompanied by reports that both *Excel* and *Works* were also being readied for the UNIX world. Evidently Microsoft woke up to the significant incursion made by *WordPerfect*'s word processors and other products into the mini-computer sphere.

Now it's OFFICE on UNIX

WordPerfect Office, its integrated office automation program is shipping for SCO Xenix. *Office* had been available for VAX/VMS environments, and *WordPerfect* had versions of *WordPerfect 4.2* already running in SCO Xenix and other UNIX environments. Major feature in *Office* is *WordPerfect Mail* which is an integrated E-Mail system offering password protection and other security safeguards in addition to ability to communicate with a number of systems using Ethernet TCP/IP or UUCP connections.

Major News Waits for COM-DEX/Fall 89

Software Releases?

What's new? Lots and little. Many software publishers are announcing new product versions, mostly in the .0X range and com-

posed of bug-fixes and minor enhancements. It's late enough in the season to hold off really hot, major releases for COMDEX/Fall 89.

Good Bets for COMDEX...

Ashton-Tate may finally throw in the towel and release *dBASE IV* as a number of separate products a-la Lotus' experience with 1-2-3 3.0 and 2.2, with the high-end (very expensive) SQL capable version possibly given a new name. With *dBASE IV's* current reputation, might be a good idea to start fresh.

Microsoft is expected to release *WORD for WINDOWS* with a parallel announcement of *WINDOWS/3.0* which is rumored to be fully multi-tasking and capable of breaking the 640K (768K?) barrier with a rumored new release of DOS. *WORD for WINDOWS* is really late arriving as it is and there's been a lot of speculation in the trade that the new *WINDOWS* is one of the reasons for the delay.

IBM is expected to announce a new PC-DOS version, perhaps by time this comes off the press, and rumors are flying it will be more than just a bug fix for the problematic 4.0X. Figuring in the rumors are massive industry acceptance of the DOS extenders, including IBM's endorsement of the ALL ChargeCard memory management add-on for 80286-powered PC's.

WINDOWS/DOS - DOS/WINDOWS

Speculation Rampant

Media speculation about the future of DOS and WINDOWS has been punching holes in the ionosphere in the last several weeks. Early rumors described a *WINDOWS/3.0 - DOS (75.0?)* combination that would use protected mode operation and allow huge memory utilization. Flying in the face of IBM/Microsoft's investment in OS/2-PM, this seems a little extreme at the moment.

ChargeCard and DESKview Gain Recognition

Wait a minute, though. IBM did

endorse (agree to sell) the All ChargeCard through its system(s). And, the ChargeCard (long a secret of BBS sysops and other mythical PC cognoscenti) does do for 80286 memory management what Quarterdeck's *DESKview* did for multi-tasking or task switching under DOS, and what IBM/Microsoft either couldn't, or wouldn't do. Quarterdeck, remember the patents unveiled in June, has long been a staple of knowledgeable power users.

Borland Votes for VROOMM and WINDOWS

Clouding the issue(s) are Microsoft's lack of comments regarding plans for *WINDOWS*. Leaks indicate that lots of developers looking at OS/2 have said they prefer working with *WINDOWS* and see more future there than with OS/2. That includes Borland whose new *Quattro Pro* is reported to outdo 1-2-3's version 3.0, and on PC/XT's no less! Borland reports they are doing it with VROOMM - Virtual Real-Time Object-Oriented Memory Manager which allows running in a 512K environment.

IBM/Microsoft Select AGFA

IBM and Microsoft are expected to incorporate Agfa Compugraphic Division's "Intellifont" outline fonts in coming versions of OS/2's *Presentation Manager* (PM.) According to some media sources, this is via Hewlett-Packard and is both a result of the dominance of HP Laserjet printers in the PC camp and HP's *New-Wave* approach. Many users had hoped for *PostScript* as a PM (and *WINDOWS*) standard, but tide seems to be turning against *PostScript* for the present as a result of Adobe's very high licensing fees.

Printed Word (and Graphic) is King

At the High End...

Laser printers at both the low and high ends of the spectrum have been making waves over the last month. Hewlett-Packard, Toshiba and Epson are all expected to an-

nounce new, low-end models in the \$1000 (real retail) price range with 4 to 6 page/per/minute outputs.

Epson Announces 6 PPM Model

Epson announced the EPL-6000 printer with 300 DPI resolution, 6 PPM output and base 512K RAM. The new Epson printer uses a Motorola 6800 CPU and can be upgraded to 4.5 Mbytes of RAM. HP graphics (HPGL) are supported and Epson suggests a \$1899 price for the new printer.

In another announcement, Epson America reported that they had signed an agreement with QMS Inc., to provide an inexpensive, software-based *PostScript* emulation throughout its entire printer line according to *INFOWORLD*. The trade journal stated that both dot-matrix and laser printers would be included in the arrangement, and that this would bring *PostScript* capabilities to PC users who couldn't afford the standard before.

Canon SX Powers 1000 X 400 DPI Printer

A Canon SX second-generation engine powers the newly released Trendsetter 1000 Plain Paper Typesetter by Raster Devices Direct, Inc. Boasting 1000 X 400 DPI resolution, the new printer is *PostScript* compatible, has 8 PPM output capabilities and comes with 6 Mbytes RAM as standard. It also comes equipped with 35 Bitstream Fontware scalable fonts, drivers for GEM, Microsoft *WINDOWS*, *WordPerfect 5.0* and Microsoft *WORD*. Announced price is \$5,995.

Lasermaster Also Boasts High Resolution

Lasermaster 1000 is a high-resolution laser printer aimed at the upper end of the DTP spectrum with 1000 X 400 DPI capability and interfaces to both the PC and NuBus (Macintosh) machines. The new printer also comes with 35 Bitstream scalable font outlines said to be compatible with the Apple LaserWriter Plus. Few technical details were available, but ►

starting price was pegged at \$7495 for PC's and \$7995 for PS/2 and Macintoshes.

Corporate Follies, 1989 version
And, We Can't Resist...

Tandy is the parent of RadioShack. That RadioShack has been successful for Tandy is an understatement of massive proportions. But, to Tandy's eyes (corporate), RadioShack has always been associated with electronics buffs, scruffy experimenters and other non-corporate types. Description covers a lot of folks.

Sometime back, Tandy announced that it was divorcing its computer operations from RadioShack and bringing them under the Tandy name which Tandy thought would appeal more to corporate buyers. Evidently that idea either didn't work as planned or has been superseded since Tandy announced in September that all single-unit computer sales would again be through the RadioShack chain.

Corporate sales will now be through Tandy's subsidiary, GRID. Probably a good idea as GRID folks tend to know their products.

Can't say the same for my experience with Tandy salespeople.

An Wang Rides Again!

Pre-PC's, Wang Laboratories was the dominant influence in corporate word processing. Wang was also a major player in business-oriented minicomputers, ranking with DEC and IBM. Much of this was attributed to the genius of An Wang, founder of Wang, whose son, Fred Wang, has been running the company of late. That is, Fred ran Wang until Wang announced a \$370 million loss during the past quarter.

Wang has had a mixed history since the PC cut the feet off the dedicated word processing market. Notoriously independent, Wang released their own PC's in 1982 which were 8086 powered, but mostly incompatible with the PC-DOS standards set by IBM. Recognizing they had seriously misjudged the market, Wang tried again with promised "full compatibility" PC's in 1986. This round was closer — about 75% compatible, but had lots of quirks and "surprizes" for users trying to run non-Wang software.

An Wang is, at the helm of Wang again, and, you guessed it, Wang has recently announced a new line of PC's that includes 80386SX and full 80386 models. Industry watchers are hoping it won't be a repetition of the last Wang PC experiments, and that it won't be too late.

While Peter Norton Juggles...

Peter Norton, best known for his *Norton Utilities*, his superb writing and wide, red neckties in old pictures, is rumored ready to release a new hard-disk backup pro-

gram "later this year." (COMDEX/Fall 89?)

That's interesting. He'll be going head-to-head with Fifth Generation Systems (*FastBack*) and Central Point Software (*PC Tools*), his most aggressive competitors. Fifth Generation recently acquired the Mace Utilities and is determined to expand that business considerably.

Adding to this, his company, Peter Norton Computing, seems to be going through a particularly painful reorganization. The last month has seen media accounts of hirings, firings, "resignations" and other personnel changes sufficient to keep a Byzantine politician happy.

Rising from the Ashes?

Phoenix Technologies is another "old," PC-built company experiencing reorganizational turmoil. Phoenix's BIOS ROMs made many early clone PC's "legal" and are still considered the best alternative available to "real" IBM BIOS ROM's by many users. (Phoenix and Award are the two dominant BIOS sets found in non-IBM, non Compaq PC's)

According to recent media coverage, Phoenix "spread itself too thin" and focused on non-BIOS business activities that "haven't done that well", including MCA technology and UNIX software compatibility projects.

As C&T Swallows SMS...

Scientific Micro Systems (SMS), maker of hard disk controllers, was acquired by Chips and Technologies (C&T) in August. SMS had been one of the leading controller developers for PC's and were considered major competition for Western Digital in the PC hard-disk controller market at the single-unit sales level.

Chips and Technologies stated that they planned to incorporate the SMS into its own mass storage and VLSI system level operations according to a recent *INFOWORLD* article, but didn't reveal plans to continue participation in the single-unit (retail) market. a

SWAP SHOP


Four lines free each month to members; 5th through 10th lines at 30 cents per word. Larger ads at commercial space rates. Send check to the Editor for words exceeding the four-line limit. Free ads are on a space-available basis.

Programmer Wanted

Degreed programmer needed with real-time programming experience and excellent low-level software design skills. Qualified candidates will have background in one or more of the following areas: MS DOS programming, LANs & data communications, and firmware development. Must be proficient in assembly language. Knowledge of C language a plus. Preference given to persons with 2 or more years professional experience. Send resume to:

Engineering Department,
Attn: MB,
10495 Olympic Drive, #102,
Dallas, TX 75220

If you want better word processing, don't settle for Perfect.



REPORT TO THE FIELD

Shoe enough
Arbor Footwear counts and counts: Jim Dearing becomes V.P. Design for Arbor Evening and Dress wear. Greg Vrooman moves from Peoria to marketing in Miami. Joan MacArthur and Lisa Doan become account supervisor and media planner, respectively. *J. Miller*


Sales per region
Across the board it looks like the South once again led sales in FY '88. With the continuing pump crank those figures should maintain, if not improve. *Ann W.* It's important not to let the other areas such as Northeast and Midwest fail.

Sales per region

Style	Dress	Sport	Casual	Costly
Evening and	Weekender	Tailor made	Trouser up	
Midnight overcoat	Kickaround	Ladies wedge	Loafers	
Twilight window	Runner up	M.B.A.-brand	Special spectator	

Price: \$150-\$195 185-\$125 \$120-\$250 \$90-\$150

Young professionals use Arbor Walker to commute.
Soon, young professional women will no longer be using Arbor shoes simply to climb the corporate ladder. They'll be using them to get there.
This fall we're introducing the Arbor Walker — a shoe that goes quite a few steps further than the now-accepted but unattractive tennis shoes women have been wearing to and from work.
Made of Italian leather with a rubber sole, the Arbor Walker is the perfect combination of style and durability.
Look for promotional materials and carrying cases in early September. Our sales point is: "Women don't have to sacrifice looks for comfort."



Arbor Walker designed by Pier Dasso.

SALES FORECAST FOR 1989

	Types of shoes			
	1st quarter	2nd quarter	3rd quarter	4th quarter
Dress	95,000	87,000	120,000	60,000
Casual	60,000	82,000	60,000	90,000
Sport	120,000	85,000	75,000	80,000
TOTAL	390,000	314,000	345,000	317,000

Issue # 022388, 4:32 PM
Let's discuss forecast for media department expansion.
Date # 07/89, 1:15 PM
Figures for Northeast and Midwest should be calculated separately next month
Let's catch them before they fall with promo.

The trouble with WordPerfect® is, sometimes it isn't.
Not when you compare it to our new Microsoft® Word version 5.0. Not when you consider that with Word 5.0, you'll be able to pump out your day-to-day jobs quicker, easier, even smarter than you ever thought possible.

For example: With Word 5.0 you can choose commands by either using simple speed keys or just selecting from a menu.

You can create documents with about 50% fewer keystrokes than you-know-who. And now enjoy even more true WYSIWYG (What You See Is What You Get) thanks to our Print Preview.



Introducing new Microsoft Word 5.0. When Perfect just isn't good enough.

Does WordPerfect give you the freedom to make annotations? Nobody's perfect. Does WordPerfect feature Outline View? Another imperfection.

New Word 5.0 also lets you list, search and archive files across multiple directories. And even link part, or all, of a Lotus® 1-2-3, Microsoft Excel or Microsoft Works spreadsheet into a Word document. And then update it. All in a matter of seconds.

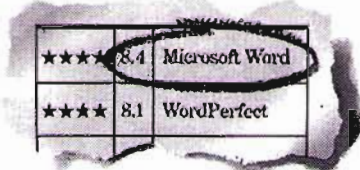
For a thoroughly convincing demo, first call (800) 541-1261, Dept. J63. Then call WordPerfect for a demo. Then run both, our Word against theirs, and then ask yourself:

Which one brings your word processing closer to perfection?

Text, graphics, fonts, annotations. Faster, simpler, cleaner. Compliments of new Word 5.0.

COMMON WORDPROCESSING TASKS		
Which requires fewer keystrokes?	Microsoft Word 5.0	WordPerfect 5.0
Copy Block	4	7
Delete Line	2	4
Italicize Word	2	5
Change Font and Size	6	9
Add Footer	1	7
Box Paragraph	5	12
Total	20	44

With new Word 5.0 you'll do about 50% less hunting and pecking than with WordPerfect.



Software Digest recently chose the best overall advanced word processor on the market. Who won? In a word, Word.

Microsoft
Making it all make sense.

Output printed by the Hewlett-Packard® LaserJet Series II® printer. Customers in Canada, call (416) 673-7638. Outside North America, call (206) 882-8661. © Copyright 1989 Microsoft Corporation. All rights reserved. Microsoft, MS-DOS and the Microsoft logo are registered trademarks, and Making it all make sense is a trademark of Microsoft Corporation. Hewlett-Packard and LaserJet Series II are registered trademarks of Hewlett-Packard Company. Lotus and 1-2-3 are registered trademarks of Lotus Development Corporation. WordPerfect is a registered trademark of WordPerfect Corporation. Software Digest rating from April 1989 issue. This offer valid only in the 50 United States.



Computer Council of Dallas
1950 Stemmons Freeway, Box 277, Dallas, TX 75207

Number 2 - October 1989. By: The Journal group.

Topics: A magical mystery tour, part one.

WHO IS THE COMPUTER COUNCIL?

We invite you to meet the CCD. Just who are they?

In short: a coalition of independent computer user-groups. By our last count, sixteen groups are currently affiliated. If you haven't walked in on a few, you've missed a lot. This month, let's start an "outsider's tour". Some general impressions and opinions, gleaned from our roving reporters:

C-CAD = Center for Computer Assistance to the Disabled

The C-CAD organization exists to help disabled persons in life and career with the aid of computers and computer training. They do all their non-profit work with money and equipment donations (NO government money!). They have placed well over 500 computers with their "clients", including special adaptations and software to suit the needs of the user. They bring some demo equipment at meetings. The staff is very dedicated. New doors of independence open to people because of their efforts; the kind of results that fuel the whole user community. We're proud they're our newest member group.

DALACE = Dallas Atari Computer Enthusiasts

DALACE, in on the CCD's birth, has over 100 devoted members. They offer diverse support, primarily for the many ubiquitous home-oriented Atari machines. The ST camp here is small but has a great deal of vigor. The DALACE heart, though, is in its multitudes. Atari sold (and still sells) boatloads of console-type computers, such as the 400, 800, 130XE, etc. Each has sophisticated video capabilities and easy-to-use facilities. As a result, applications including graphics and sound get particular attention; note that there are serious gamers here. Also plenty of good general discussion and BASIC instruction; many interesting presentations. All computer users are welcome here. Beginners and home users especially will appreciate the generous flow of creative ideas.

TIPRO = Texas Instruments Professional Computer Users

This 200-member group caters mainly to the users of the Intel based TI products such as the Professional and TI Portable Professional. But the talk is often much more generic and MS-DOS

oriented (the TI machines are very close to XT's-- and better than them in some ways). Meetings are professional affairs, well-organized. Extensive disk catalog. Business applications, of course, predominate; these guys are very well-versed in all hardware aspects as well. The group takes care of its own, with ribbon-reinking and other valuable services to its members. TIPRO has also provided many man-hours for projects in the community. Whatever you do, don't miss their annual auction!

So far we've only covered three of the sixteen. Stay tuned, for we will continue next month. Should any of these groups sound like they practice your kind of fun, we suggest you find them for yourself. It is really a good experience to see other users and what they do. That's one of the CCD fundamentals: the interaction of users benefits all groups.

We are proud of the high caliber of our affiliate groups. Take your own tour-- you'll be pleased.

Any feedback, comments or suggestions should be forwarded to:
CCD Publicity
1950 Stemmons Freeway, Box 277
Dallas TX 75207.

(c)1989 CCD Publicity Committee

**North Texas PC Users Group
Personal Users (Beginners) Special Interest Group**

**"Fundamentals of Personal Computers"
16-Class Revolving Schedule**

Inform Saturday	Class Number	Class Title/Description
12 Aug 89	1.2	Start Up
	2.2	Diskette Sizes & Formatting Each
	3.2	Copying & Backing up Files
	4.1	Personal Computer Hardware
23 Sep 89	5.1	Fixed Disk Directories, Batches, & Paths
	6.0	DOS Menu Systems on Fixed Disks
	7.1	Fundamentals of Lotus 123
	8.2	Fundamentals of "BASIC" Language
14 Oct 89	9.1	Geneals & Overview of Computer Languages
	10.4	NTPCUG Disk of the Month Library
	11.1	PC Graphics Modes
	12.1	Bulletin Boards & Archive Programs
11 Nov 89	13.0	Printer Setup
	14.0	Writing Lotus MACROS
	15.0	Major Categories of Software Available Today
	16.0	PCs to the End of the 20th & Into the 21st Century

Classes are free and open to all beginners, novices, new PC owners, soon-to-be PC owners, and personal (vs. professional) users. COME JOIN US AS WE COVER THE FUNDAMENTALS!

ON COMPLEXITY

No. 32 in a Series

Jim Hoisington

While trying to install a long awaited release of a popular software package, I got the following message, "Error loading driver file LI3PCOMN." Somewhat later I discovered that the meaning of the message was "Out of memory."

One of the problems with computers has always been their poor methods of communicating error conditions to the humans operating them. The tabulating machines that I first programmed had no error indicators. They just stopped. It was up to the programmer to see which relays were set in response to the card just read and then to re-wire the program board. (Programming was done with wires plugged into holes in large panels.)

The early computers had lights that showed the status of the internal registers in the CPU. On the earliest machines, I remember watching the program run, instruction by instruction. To communicate errors to the humans operating the machines, the programmer would load a certain register, usually a register called the accumulator, with a pattern of lights to indicate a particular error condition and then stop the execution of the program with that pattern displayed in the lights.

One time I was in a staff meeting in my boss's office when an urgent call came in from our plant in New Jersey. They insisted that it was an emergency, so the secretary interrupted our meeting and my boss put the call through on his speaker phone.

It seems that their computer was not working at all. They were sure that the computer was totally destroyed and insisted that I catch the next plane from Houston to supervise the repair or replacement of the machine. (In their distress, I am sure that they overlooked the fact that they were playing to an audience of those assembled in our staff meeting over the speaker phone.)

I asked them if the console lights were on and they assured me that they were. I then had them read out the accumulator lights even though they told me that it was useless because they had already tried everything and the computer was hopelessly broken. After determining that the accumulator code was 4002 (hexadecimal), I asked the operator to walk over to the printer, power it on, and to push the START button on the computer. The phone connection was broken

shortly after we heard the sound of the printer printing the job information for the first job it had processed.

Mainframe computers have not progressed much beyond the register lights scheme. They typically give the user a message code and possible one or two words as a clue to the meaning of the code. It is up to the user to find a more complete, but highly technical, explanation of the code in one of the several manuals containing the error codes for that system. Sometimes, the most sought after expertise on a mainframe in the ability to interpret error messages like, "IEH403I - Data Exception."

Error messages have always been a problem for programmers because the cause for the error is obvious to them at the time they are putting together their program. So obvious, in fact, that they are sure that it would need only the briefest of explanations. The problem is that the problem is almost never obvious to the user and needs to be explained in detail.

Personal Computers increasingly do not require that we have an expert or 3 thick manuals full of error codes present to interpret what went wrong. Instead, on-line, context sensitive help screens are being increasingly used on Personal Computers to inform the user exactly why the computer is having trouble. As an example, when using the database Advanced Revelation, I can optionally use a Concept Help, a Specific Help, an Edit Specific Help, and a screen showing all possible entries called a Popup for each field on the screen. The idea is that the computer should be able to answer most, if not all, of the questions a user might have about the purpose of the field and what kind of data can be properly entered into it.

Occasionally programmers put error messages into programs that indicate that all is lost. The programmer knows that this particular error condition should never happen, but if it does, the user should stop immediately and call for help. These kind of messages tend to get past most project managers and never make into the documentation.

The funniest error message that I have heard of comes from Ken Land. Once, he was testing a piece of software and deliberately trying to make the it malfunction. In one instance, the package came up with the following message just before it locked up the whole machine - "Sub-Queue processing approaching non-linearity!" Now that's impressive!

Jim

▲



Inside the North Texas PC Users Group Community

Connie Andrews, Volunteer Coordinator
 Andy Oliver, Assistant Volunteer Coordinator

Our Program Chairman, Timothy Carmichael, spends a good deal of time each month contacting vendors and scheduling and confirming their presentations in the auditorium at Infomart.

Then comes meeting day. You've probably noticed Timothy and Tom Fowlston in the auditorium setting up audio/visual equipment and auditorium lighting. Tom has been doing this since our meeting days back at Jesuit High School. Along the way, they've encountered some crazy situations.

Like the time a vendor managed to lock himself out of his house, presentation and keys inside, with very little time left to get to Infomart for the setup. Somehow he retrieved his materials and came racing in with only moments to spare, but then discovered he didn't know how to install his own software! As the vendor began his talk, Timothy, who had

never seen the program, managed to load it and pull everything together. Goodshow!

A lot of work and very efficiently done over the years, Timothy and Tom. Thank you for your dedication.

One of the benefits of NTPCUG membership is the drawings for members only at the monthly presentations in the auditorium. Club policy is that volunteers scheduled and on duty at the time of a drawing on meeting day are eligible to win even though not in the Auditorium.

In this issue we are acknowledging volunteers who served for the month of August. Our officers, directors, SIG coordinators and leaders, newsletter publisher, editor, staff and writers, and BBS staff are all volunteers; their names are listed in other sections of this newsletter.

INFOMART Liaison
 Stuart Yarus
 Robert Hilliard
 Bob Russell
 Archie Pinkney

Presentation/Equipment Setup and Breakdown
 Timothy Carmichael
 Tom Fowlston
 Pamela Brown

Information/Registration Booth
 Connie Andrews (Anchor)
 K. B. Barton (Anchor)
 Lonnie Cordell (Anchor)
 Eta E. Eta
 John Ferguson (Anchor)
 Paul Fredd (Anchor)
 Ron Gonzales
 Rick Griffith (Anchor)

Allan Harbough (Anchor)
 John Mackoy (Anchor)
 Claude McClure
 Andy Oliver (Anchor)
 Neal Rice
 Robert Richmond
 Douglas Scott
 Revis Smith
 Andrine Stricherz (Anchor)
 Connie Testa (Statistician)
 Paul Williams (Anchor)
 Betty Wright

Disk of the Month (DOM):

DOM Table
 Not available at press time

DOM Volunteer Coordinator
 Bill Drissel

DOM Central Committee
 Not available at press time

DOM Review/Presentation
 Not available at press time

Bulletin Board System (BBS):

BBS Sysops
 Tom Prickett
 Maggie Moomey

BBS Steering Committee
 Andrew Chalk
 Kent Cobb
 David McGehee
 Pete Testa
 Fred Williams

BBS Champion
 Steve Fleming

Newsletter Exchange
 PehL Lee

Public Relations Committee
 Francis Bright
 Annette Hyde
 Peh L. Lee
 Elwood Lindell
 Charles Lucas
 Tony Noguerras
 Reagan Andrews

PSSSSS!
GREAT BUNCH!



VOLUNTEER INFORMATION

1. Via BBS: (817) 461-0425 (metro) or (817) 461-0506 (metro). Sign up on the Volunteer Conference - make the subject matter your area of interest.

2. Meeting day: Sign up at the Information Booth or DOM Booth to work those areas in a coming month.

3. By phone:

Auditorium Presentations	
Timothy Carmichael	331-6303 (h) 661-4626 (w)
DOM Booth Activities	
Bill Drissel	264-9680 (h)
DOM Software Review	
Howard Hamilton	644-5721 (h)
Information Booth and General Information	
Connie Andrews	828-0699 (h)

How You Can Convert Your XT Into a Stealth AT

Zack Porterfield

How did we, and how can you, get AT processing speed from your XT? Our problem (our being Paul Fredd and Zack Porterfield), was to achieve reasonably fast AT processing speeds (10 or 12 MHz) from XT nodes running on a Novell network (EIS II). The approach taken, and processing enhancements achieved, are applicable to the single user system as well.

There are at least two ways of "converting" your XT to a stealth AT:

1. Install an expansion board on the XT bus which contains an 80286 processor.
2. Install a new mother board with an 80286 processor.

We chose replacement of the mother board over the expansion board since this provides your former XT with the AT bus. This second approach is the one we will discuss.

There are many "hybrid" mother boards available to provide the conversion. We consider them hybrids in that they have a physical board layout and numerous mounting holes with which to accommodate installation into an eight slot XT clone case (we have not installed these mother boards in a five slot PC). Of the two replacement mother boards used it was possible to retain use of the XT case, monochrome monitor and video board. To date, we have replaced mother boards in six different XT clones and have had no expansion board alignment problems. In all six conversions, the expansion slots in the mother board matched up nicely to the slot openings in the back plate of the XT case.

The first mother board we tried provided 10 MHz speed and allowed continued use of all XT components such as XT only keyboards, XT bus floppy controller, hard disk controller, video board and slow memory chips (150ns and up) in the 64kb and 256kb DIP configuration. The second, and preferred by us, mother board was manufactured by NIC.

The NIC AT286 mother board provides 12 MHz speed but several caveats must be addressed.

* **Keyboard.** You must have an AT compatible keyboard. In many cases this is not a problem because several years ago clone makers began to build in AT/XT mode switches on the bottom of keyboards. We found that 5 out of 6 keyboards had the AT/XT switch and, once switched to AT mode, only one failed to

operate with the NIC mother board during installation check out.

* **Diskette Drive Controller.** Most older XT clones have a diskette drive controller designed for the 8 bit bus. If this is the case then you must upgrade this controller board to an AT level diskette controller. Of course, try the old controller first but be advised that this requirement was true in all but one of our NIC mother board installations.

* **Video Board.** We converted one XT whose full length monochrome board (circa 1983) would not drive the monitor with the NIC mother board installed. This video board was in working condition before the conversion. We experienced no problems with CGA boards, regardless of the vintage encountered. None of the systems we converted had EGA or VGA video.

* **Memory Chips.** For all but one of our NIC conversions we were able to utilize the original DIP memory chips from the XT mother board. A total of 8 modes are available through DIP switch settings which vary depending on the memory sizes you plan to utilize. Both 16 and 18 pin DIPS are accommodated. Up to 4 megabytes may be installed on the mother board. NIC recommends that you set a jumper on the AT286 mother board to one wait state if you use chips with 150ns. NIC states that you can set this jumper to zero wait states if you have 100ns or faster memory.

* **Power Supply.** XT clones containing from 135 watt to 200 watt power supplies were upgraded without failure. However, in no instance did we add features, for example, a second hard disk drive, on those XT clones with smaller power supplies.

* **Hard Disk Controller.** In all cases we were able to utilize the existing hard disk controller with installation of the AT286 mother board. And we did not have to reformat the hard drive!

For those of you who need expanded memory for now older versions of Lotus, NIC includes with each 286 mother board a copy of EMS ver 4.0 from Sun Electronics Corporation.

The results of your conversion efforts may differ from ours so be prepared to purchase a new driver board here and a new set of memory chips there. The results? A Norton SI performance rating of 11.6!! A 12 minute recalculation of a Lotus spreadsheet was reduced to just under 4 minutes. Using RightWriter on a WordPerfect document the grammar/punctuation check out was reduced from nine (9) minutes to three (3).

Zack

▲

MEMBERSHIP APPLICATION

North Texas PC Users Group

The NTPCUG is a non-profit, independent organization of individuals learning to apply personal computers to practical problems. For additional information, call (214) 746-4699.

Member # _____

Name: (Last) _____ (First) _____

Address: _____ (Suite/Apt) _____

City: _____ State: _____ Zip: _____ - _____

Phone: (Check Preferred No.) Home () _____ Metro? Y N
 Work () _____ Ext. _____ Metro? Y N

Occupation/Profession: _____

Check one from <i>each</i> column below:		
Payment:	Membership Classification:	Application Status:
Cash <input type="checkbox"/>	Regular (\$24.00) <input type="checkbox"/>	New Member <input type="checkbox"/>
Check <input type="checkbox"/>	Student (\$12.00) <input type="checkbox"/>	Renewal <input type="checkbox"/>
Credit Card <input type="checkbox"/>	(full-time with ID) <input type="checkbox"/>	Address Change <input type="checkbox"/>

Do you want access to the NTPCUG Electronic Bulletin Board? Y N Already Have

Please initial here _____ if you do not wish to have your address included on member lists sold for the NTPCUG's benefit to advertisers of IBM compatible products.

The NTPCUG expects and encourages volunteer participation by members in assisting with monthly meetings at INFOMART and other activities during the month. This usually consists of a few hours of your time each year. If asked, would you consider working with one or more of the following activities:

1. Working with NTPCUG Volunteer Committees? (Please check all that apply.)

- | | |
|--|--|
| <input type="checkbox"/> Bulletin Board (BBS) | <input type="checkbox"/> Information/Registration/Membership |
| <input type="checkbox"/> Disk of the Month (DOM) | <input type="checkbox"/> Newsletter |
| <input type="checkbox"/> Equipment Setup/Breakdown | <input type="checkbox"/> Public Relations/Advertising |
| <input type="checkbox"/> Financial/Bookkeeping | <input type="checkbox"/> Special Interest Group Coordination |
| <input type="checkbox"/> INFOMART/Vendor Setup | |

2. Working with Special Interest Groups? (Please check all that apply.)

- | | | | |
|---|--|---|---|
| <input type="checkbox"/> Astrometry | <input type="checkbox"/> Assembler | <input type="checkbox"/> Business Applic. | <input type="checkbox"/> C Language |
| <input type="checkbox"/> Communications | <input type="checkbox"/> Cryptanalysis | <input type="checkbox"/> DAC Software | <input type="checkbox"/> DBase |
| <input type="checkbox"/> MS/PC-DOS | <input type="checkbox"/> Genealogy | <input type="checkbox"/> Graphics | <input type="checkbox"/> Hardware Solutions |
| <input type="checkbox"/> Local Area Net | <input type="checkbox"/> LOTUS | <input type="checkbox"/> Personal Users | <input type="checkbox"/> Advanced Programmers |
| <input type="checkbox"/> R:Base | <input type="checkbox"/> Stock Market | <input type="checkbox"/> Turbo Pascal | <input type="checkbox"/> WordStar |
| <input type="checkbox"/> Microsoft WORD | | | |

3. Being a volunteer, informal "consultant" in your area of expertise for NTPCUG members?

If so, list area(s): _____

Detach below for receipt.

Applications should be mailed to: North Texas PC Users Group, Inc.
 P.O. Box 780066
 Dallas, TX 75378-0066

Received: \$ _____ Check No. _____ Date: _____ / _____ / _____ By _____

Meetings & Times



9:00 AM - 10:00 AM

Syscorp International

MicroSTEP

Speaker: Thomas Meadow, Senior VP

10:00 AM - 11:00 AM

Good Software Corporation

Arriba - The Versatile Personal Information Manager

Speaker: Elliott Lowe, Director of Marketing

(See page 1 for description of programs.)

Business Meeting
11:00 - 11:25

1:00 PM - 2:00 PM

Prodigy Services

Prodigy Services - New Dynamics In Interactive Online Services

Speaker: Jan Edzards, Sales Rep.

Special Interest Group Meetings

For possible time changes, check the Bulletin Board just before the meeting and the overhead display in the lobby at INFOMART.

9:00 - 9:55	10:00 - 10:55 (cont)	12:00 - 12:55 (cont)	2:00 - 2:55
Assembler Community Volunteers DOS Hardware Solutions Personal Users	Local Area Networks Personal Users	Personal Users RBase Stock Mkt Investing	Advanced Programmers Cryptanalysis DAC Easy Accounting Databases
<u>10:00 - 10:55</u>	<u>11:30 - 11:55</u>	<u>1:00 - 1:55</u>	
Astrometry Graphics	Orientation	Business Applications LOTUS Personal Users Turbo Pascal WORD	
	<u>12:00 - 12:55</u>		
	C Language Communications		



North Texas PC Users Group, Inc.

P.O. Box 780066, Dallas, TX 75378-0066

Phone (214)746-4699 for recorded information about the User Group and meeting dates.

The North Texas PC Users Group, Inc., is a non-profit, independent group, not associated with IBM or any other Corporation. Membership is open to owners and others interested in exchanging ideas, information, hardware, predictions, and other items related to IBM Personal and compatible computers. To join the Group, complete the application blank printed elsewhere in this newsletter, and send it with \$24 membership dues to the Membership Director whose address is shown below. A subscription to the newsletter is included with each membership. The Group meets once each month, usually on the second Saturday. See cover for date, time and place of the next User Group meeting.

Board of Directors

Jim Holsington, Phil Chamberlain
Chairman Sid Nolte, Ph.D.
Reagan Andrews, Ph.D. Zack Porterfield

Officers

President Jim Holsington (214)416-3101 h
President-Elect Zack Porterfield (214)931-4426
Program Chair Timothy Carmichael (214)331-6903 w
Treasurer Ken Conner, CPA (214)669-3377 w
Secretary David McGehee (214)681-0202 h
Membership Dir. Jim Holsington (214)416-3101h
Advertising Dir. - OPEN -
Disk of the Month Kathryn Loutman (214)596-2539
Group Statistician Connie Testa
Volunteer Coord. Conrie Andrews (214)828-0699

Member Emeritus

Stuan Yans

NOTE: To access the BBS from outside Area Code 817, use Area Code 817. (This is NOT a toll call from Area Code 214.)

BULLETIN BOARD (817)461-0425 (Metro)
(817)461-0506 (Metro)
SYSOP: - Tom Prickett (voice) (214)690-9067
Asst. SYSOP: - Maggie Mooney
Technical Advisors: Fred Williams
Pete Testa
User Relations: Kent Cobb
Information Mgt: Dan Marmion
Technical Services: Dwight Neal

Address Changes, etc...

Payment of dues, address changes, and inquiries about membership should be directed to

NTPCUG Membership Director
P.O. Box 780066
Dallas, Texas 75378-0066

(Check newsletter mailing label for your renewal date.)

Special Interest Groups

SIG Coordinator	Zack Porterfield	(214)434-1844 w
	Phil Chamberlain	(214)243-5034h
Astrometry	Arin Collins	(214)351-5137 h
Assembler	Andrew Chak, Ph.D.	(214)226-3461 h
Business Applic.	Bruce Schubert	(214)348-5700 w
C Language	Sid Nolte, Ph.D.	(214)233-8178 h
	Andrew Chak, Ph.D.	(214)226-3461 h
	Stan Miam	(817)548-1573
Communications	Pete Testa	(214)496-7506
	Wm. Bennett	(817)346-0862 h
		(817)762-9059 w
Cryptanalysis	John Taber	Metro 430-8173
	John Thomas	(214)860-1823
DAC Software	Greg Cohen	()
Databases	Rodney Haas	(214)255-4400 h
		(214)404-4812 w
DOS	Jim Holsington	(214)416-3101 h
	Reagan Andrews, Ph.D.	(214)828-0699 h
Genealogy	Minnie Champ	(214)644-8643 h
Graphics	Richard Terro	(214)307-1259 h
How Solutions	David McGehee	(214)681-0202 h
	Gary Johnson	(214)937-9676 w
		(214)937-5851 h
Local Area Net	Fred Williams	(214)492-1315
	Dan Marmion	(214)750-6130
LOTUS	Mark Gruner	(214)964-8174 h
	Pat Henley	(214)229-9218 h
Personal Users	Bob Presley	(214)867-1879 h
Programmers	Kent Cobb	(214)343-3554
	Jim Holsington	(214)416-3101 h
	Alan Alberts	(214)242-1094 w
RBase	Con Branham	(214)352-0868 h
	Cliff Murphy	(214)279-7973
Stock Market	Richard Holerman	(214)341-4774 w
	Don Chick	(214)276-2524 h
Turbo Pascal	Jay Shilstone	(214)627-5751 h
Volunteers		(214)361-9681 w
		(214)750-6130
WORD	Dan Marmion	(214)828-0699
	Reagan Andrews, Ph.D.	(214)681-0202 h
	David McGehee	(214)681-0202 h
	Dorothy Berlina	(817)367-9983 h



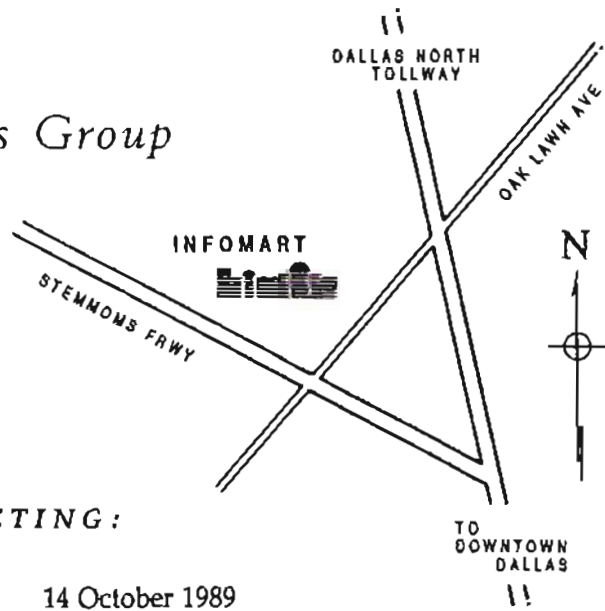
North Texas PC Users Group
P.O. Box 780066
Dallas, TX 75378-0066

Non Profit Org.
U. S. Postage
Paid
Arlington, TX
Permit No. 823

Address Correction Requested.



North Texas PC Users Group



NEXT MEETING:

14 October 1989