OPCUG The Ottawa PC User's Group

Mailing Address - 3 Thatcher Street, Nepean, Ontario K2G 1S6
(613) 723-1329 (voice) (613) 723-7064 (BBS)

May 1990 Vol. 7 No. 5

LIES, DAMN LIES AND STATISTICS

by John Whelan

If we look at users of the OPCUG's Bulletin Board, the PUB, they like to call in whenever they like without getting a busy signal. Most members log on during weekday evenings around 8 p.m. A few call at 2 p.m. and fewer still at 2 a.m. If they get one busy signal for every 20 dialling attempts, they are satisfied. This should perhaps be our ideal level of service.

If users wish to read the opening screens and menus and check the Buy & Sell and General Message areas, on-line sessions take 100 seconds. If messages are read on-line, sessions take longer.

ROBABILITY THEORY & STATISTICS

Probability theory and Statistics are rather fun and useful branches of mathematics that are often maligned and misunderstood. However, they are useful in studying bulletin boards. Using Erlang (a copy awaits you in the file section), we can build a mathematical model of a BBS. If it has one phone line, we find we can support three callers per hour, given the above characteristics. If we increase the number of lines to four, we can support 49 callers per hour. In practice, this means members can usually get through to the PUB.

OPTIONS

The PUB has become more and more popular with members. If Carl-Henri's "Member-Get-A-Member Relay" plan succeeds and the Group gets numerous new members, what options are at our disposal that will keep users satisfied?

We could start an advertising campaign in the Newsletter asking members to log on to the PUB between the hours of 2 a.m. and 4 a.m. rather than between 7 p.m. and 10 p.m. I'm sure Bonnie would rise to the occasion, but whether she would be prepared to get up at such an unearthly hour, I wouldn't like to say.

We could add a fifth line which would allow 69 callers per hour, but it would cost the Group an additional $400 or more per year plus over $200 for another modem.

JOHNPUB

There is another alternative: get members to use an automated system called JOHNPUB. With this system, just logging onto the PUB to check for messages takes only 22 seconds of connect time. This would allow 220 callers per hour with the current four lines.

With JOHNPUB, messages can be read and composed off-line saving several minutes of connect time. Downloads terminate on-line sessions immediately after their completion, saving another two minutes of connect time. Thus, the number of members who log on just to download a 10-minute file, increases from six to eight per hour.

JOHNPUB helps with peak times because it runs unattended which frees up peak-time availability. It uses Telix's ability to run under the control of a script file and produces a system that automates the process of receiving and sending messages and downloading files.

My father is not the most technically advanced PC user. I described, via Transatlantic phone calls, how to change directories by saying, "Well, there's a key that has a line from top left to bottom right somewhere on the right-hand side..." so the keywords or commands that I created had to be simple enough for him to understand.

Bonnie did such a good job of translating the documentation into English, that I'm going to cheat here and let the documentation explain the product.

(Continued on Page 2)

NEXT MEETING

The next meeting of the Ottawa PC Users' Group will be held on May 30, 1990. The Guest Speaker will be Jamie Simser, a WordPerfect Corporation representative. His topic will be the newly released graphics package DrawPerfect".

FROM THE EDITOR

Hello fellow members. I would like to explain the purpose of the Question Box that I brought to the last meeting (if anyone could recognize it as such!). Its intent is for questions that members have but are, for one reason or another, unable to ask them at meetings. All questions that are put in the box will be answered either at the meeting itself or directed to someone who can answer them. So I'll bring an 'upgraded' box to the coming meeting. Let's see if we can fill it up at the upcoming meeting!

I would like to thank the group from The Xerox Corporation for a fine presentation of their graphics packages.

* * *

CONTENTS

Meetings are held at 100 Sussex Drive, Ottawa
Last Wednesday of the month - 8:00 p.m. to 10:00 p.m.
No meetings in July and December

JOHN'S OFF-LINE PUB SYSTEM

(Continued from Page 1)

WHAT IS IT?

JOHNPUB logs onto the PUB unattended and reads any messages it finds in the file areas that you have selected and captures them to a file called "pub1in.txt". Have the enclosed files of private messages downloaded. Private messages are deleted after reading. It sends any messages that you wish to send or have replied to and retains the reply chains.

My idea was to write a system that would allow messages to be read and captured, including enclosed files, from the PUB while the computer was unattended. These messages could then be replied to off-line, using an editor such as WordPerfect with its spellchecker. The idea came about due to Bell's charges for transatlantic phone calls and the existence of Pat Whelan's PC. It was developed from a script Morris Turpin sent to me. Various people, including Chris Taylor, provided input along the way. Bonnie Carter translated the documentation into English.

WHY BOTHER?

Well, the PUB can support many more callers with its existing equipment if calls are short and spread out throughout the day. With JOHNPUB, you can tell your machine to call the PUB in the afternoon. Then, at your leisure, you can read and reply to messages and zap the replies off with a very short connection to the PUB. You can even tell it to download a couple of files while its at it.

HOW DOES IT WORK?

It uses a file "Pub1in.txt" as the basic input and output of the system and a small number of commands starting with "#". Edit this file with any editor and save the output as ASCII text.

To reply to a message, put "#reply" at the start of a blank line below the message. End your reply with a "#" at the start of a blank line as well. Your reply will be added onto the end of the message chain so that others can use the <-> command to follow the message chain to the original message.

To send a message, use the "#send" command (i.e., #send P John Smith Subject). Leave a space between the parameters. The first parameter represents the message area. You must use a CAPITAL letter; P for private, G for General and B for Buy & Sell. The next two parameters are the first and last names of the person you are sending the message to. If you wish to send to "All", use "#" for the last name and it will be ignored. The rest of the line is for the subject. Begin your text on the next line. Terminate the message with a "#" at the start of a blank line.

To enclose files with a message, end messages with the" #encl" command. #encl fred1.zip c:\wpdoc\fred.zip attaches the file c:\wpdoc\fred.zip to the message, calling it fred1.zip. Enclosed files may be attached to messages that you reply to or send.

To pick up files, use the "#get" command. Edit the file pub1in.txt so that the line starts with #get, followed by a space, the number of the file area and the filename. For example, "#get 1 allfiles.zip" will give you a list of all the files and their areas on the PUB.

SETTING UP

To run JOHNPUB, put all the files included in this zip into a subdirectory called "c:\pub". You must have Telix in a subdirectory called "c:\telix31", but this can be customised by modifying m.bat.

You may have to change your PUB configuration. At the main menu, type U for users, then C for Config, M for message prompts (I usually select 3). You will be asked if you wish to be prompted for prepared messages. Answer N. Change your default download protocol to YModem Batch because it passes filenames across as part of the file transfer.

Pub1parm.txt contains four lines of parameters. Line 1 is your capture file, normally c:\pub\pub1in.txt. Line 2 is the dial line of a telix script file (I use {dial("1",10,1)} to dial the first number in my telix dialling directory.) Modify it to fit your directory. Change Line 3 to reflect your name, city and PUB password.

Line 4, if the first character is S, omits the PUB's opening screens. If the second character is P, the Private message area is checked. If the third character is G, the General message area is checked. If the fourth character is B, the Buy & Sell area is checked. If the fifth character is S, JOHNPUB does its work and then stops and waits while you work directly on the PUB. Here you can change configurations, upload files, capture enclosed files in the General area and do odd things that JOHNPUB can't do. Save this file as an ASCII text file. Then run m.bat by typing M <Enter>.

When using the system, customise your telix configuration a little. Since you aren't waiting in front of the machine, increase the delay between diallings to a larger number of seconds, such as 600. Telix then waits ten minutes between diallings. If you set it to dial ten times, it will make the attempts over an hour-and-a-half rather than two minutes.

"m.bat" runs pub1 twice. The second run splits the capture file up into individual files for the General (G.txt), Private (P.txt) and Buy & Sell message areas (B&S.txt). This makes message pickups clearer.

JOHNPUB looks for the first line starting with a carriage return character to spot when a message is terminated, so if someone has loaded a message in with a file transfer it ends the message a little early. However the entire capture file is still available (pub1in.out) for those wishing to look at the opening screens or see why something went wrong with the connection, or for those odd messages that don't display correctly, etc.

"M.bat" stores the messages found in each area, i.e., in P.log for Private messages, G.log for General Messages and B.log for Buy and Sell messages. This enables you to review old messages if necessary. Because these log files grow, delete them from time to time or purge some messages with an editor of your choice.

M.bat may be tailored. Use one of the delaying programs within m.bat to set off the connection at 4.30 a.m., a quiet time on the PUB. You might wish to use G.txt or the other files to merge them together.

LIMITATIONS

You are restricted from using some quotation characters in messages and from using the "#" at the start of a line.

The system was designed to run on a 4.77 mhz XT at 2400 baud. This allows you to use error-correcting modems. It may not work at a higher speed on this class of machine. For those with faster machines, higher speeds should be possible.

If someone sends you an enclosed file in the General or Buy and Sell areas, you must pick it up yourself. The messages coming back from TBBS are much too complex to allow sufficient tracks within Telix to handle all the permutations on an XT at 2400 baud.

(Continued on Page 3)

JOHN'S OFF-LINE PUB SYSTEM

(Continued from Page 2)

The Ymodem-Batch protocol does allow wild cards within the filename, so "#get 5 a*.zip" will download all the files that meet the criteria until you run out of time.

ITEMS TO NOTE

From time to time the system will appear to hesitate. If you have an external modem, you will note that the SD and RD lights continue to flash. Telix defers writing to the screen sometimes when it is busy sending to the PUB. Also, there appears to be a delay between the time the system has finished and before it hangs up, often giving you a 'Hang up failed' message. JOHNPUB sends the sequence down to the PUB to log off so that the PUB will disconnect. It also follows this after short delay with a 'Hang Up' message to the modem by which time the call should have already been terminated sometime earlier. Should the script hang at any point, the TBBS will disconnect you after two minutes without input.

Pub1 is a Pascal program that writes a Telix .slt file called "pub1.slt". The pub1.slt file that I created has not yet been fine-tuned. Any suggestions or modifications are welcome, but please edit pub1.slt and try them first. Getting this far has involved the PUB's TBBS board putting up with many high-speed running amok scripts.

Personally I keep two pub1parm.txt files one called pub1pars.txt which is used with mm.bat. The difference is that this one has the "S" in the fourth character position on the third line so that the system leaves me manually logged on after picking up and dropping off messagesy. I also have a .bat file called n.bat that contains the line "wp pub1in.txt" this brings up WordPerfect with Pub1in.txt already to read or edit.

If you have any suggestions, etc., leave me a message.

* * *

EASY RE-CONFIGURING

by Michael Goddard

The new software you installed doesn't seem to work properly. Perhaps it interacts with some of the TSRs you load in your AUTOEXEC.BAT. You must reboot your system with a bare-bones AUTOEXEC.BAT to check it out.

Then there may be times you need to use ANSI.SYS and other times you don't, and from what I hear about Lotus 3.0 (I don't have a copy yet to verify this), you may need to configure your machine specifically to run 123!

Here is a handy trick that Duncan Murdoch and I use to eliminate all that renaming and rewriting of CONFIG.SYS and AUTOEXEC.BAT.

First, create a new subdirectory and call it "BOOT". Next, create separate subdirectories off BOOT, one for each configuration you want. Each subdirectory has its own customized CONFIG.SYS and AUTOEXEC.BAT.

For example, with 384K extended memory, I have separate directories for a full 384K RAM drive, a 64K cache and 320K RAM drive combination and some other configurations too. You also want separate subdirectories for your standard configuration and for a "vanilla" configuration.

There are two files in the BOOT subdirectory itself. The first is a batch file. An overly simplified BOOT.BAT would be:

```
COPY %1\*.* C:\
BOOTER
```

and you would invoke the batch file by using the name of the subdirectory containing the configuration you wish.

The idea is that, for example, by entering "BOOT VANILLA", you would overwrite the existing C:\AUTOEXEC.BAT and C:\CONFIG.SYS with the new configuration files and then reboot your system.

When you finish testing the system with the vanilla configuration, you need only go back to the BOOT subdirectory and type "BOOT STANDARD" to go back to your usual configuration.

The second file in the BOOT directory is BOOTER.COM. The following debug script will create the file BOOTER.COM on your disk.

```
N BOOTER.COM
A 100
XORAX,AX
MOVDS,AX
CS:
CMPBYTE PTR [0080],00
JZ0114
MOVWORD PTR [0472],0000
JMP011A
MOVWORD PTR [0472],1234
JMPFFFF:0000
                <---important blank line!
RCX
1F
W
Q
```

If you type BOOTER with no command line arguments, your system will do a reboot by jumping to the code at FFFF:0000. It's a "warm" reboot if the word at 0000:0472 is 1234 (hex). If the word at 0000:0472 is 0000, then a memory test precedes the reboot. You can invoke this test by entering any character(s) on the command line, i.e. "BOOTER X".

You can do some clever things in your BOOT.BAT file. I find Chris Taylor's clever observation that the PATH variable shifts strings to upper case a real code saver here. Here is a more glorious version of BOOT.BAT.

```
ECHO OFF
IF "%1"=="" GOTO OOPS
SET OLDPATH=%PATH%
PATH=%1
IF "%PATH%"=="STAND" GOTO
  STAND
IF "%PATH%"=="VANILLA"
  GOTO VANILLA
ECHO Bad BOOT configuration!
PATH=%OLDPATH%
SET OLDPATH=
GOTO END
:STAND
COPY STAND\*.* C:\
BOOTER
:VANILLA
COPY VANILLA\*.* C:\
BOOTER
:OOPS
ECHO Give the name of the
  configuration!
:END
```

Be imaginative!

* * *

A BRIEF INTRODUCTION TO THE CONFIG.SYS FILE

by Eric Clyde

The following is based on a handout given at the 'Help Session For Beginners' at the June 1989 meeting.

When your microcomputer is booted (started), and after the system check is carried out, DOS looks for two files, CONFIG.SYS and AUTOEXEC.BAT, in the root directory of your hard disk or floppy. Most applications will run without them, but in general, your micro will run more effectively if you set these files up to suit your needs.

CONFIG.SYS was introduced in DOS 2.0. It allows you to decide how much memory to set aside to handle files, how many buffers to set aside for reading from disks, special devices (disk drives, tape drives, printers, mice, etc.), and the environment size.

The file must be in ASCII (text) format, and it can be prepared by typing the statements in at the DOS prompt level (copy con config.sys), by using a word processor (most word processors are capable of producing ASCII or text format output), or by using EdLIN (supplied on your DOS diskette).

A very simple CONFIG.SYS file would be:

FILES=20
BUFFERS=15

To find the optimum config.sys for your application, you should consult the software manuals giving the requirements for the applications you use. If these give no guidance, try experimenting. The various options are explained below, and a detailed example is given at the end.

(*) BUFFERS=N Default
is 2 or 3 (except in DOS 4.0). Each buffer uses 512K memory. The default was set when micros had much smaller memories. Unless you are short of memory or your application requests more, I recommend 10 for floppy disk systems and 15 for hard disk systems.

(*) FILES=N
specifies the number of files that may be opened at the same time. The default value is 8, but some programs require more. The maximum is 255. I recommend 20.

(*) DEVICE=[filespec]
DOS automatically loads the standard device drivers (programs which control the monitor screen, printer, diskette, hard disk and clock). If you want more control over the screen output or use special devices, this function tells DOS where to find the needed programs. Many device drivers are provided with DOS, but are not loaded automatically. If you want to use them, you must manually add them to your CONFIG.SYS file.

(*) DEVICE=ANSI.SYS
allows enhanced input and output control. It is required by many application programs and can also be used to control the colours of your screen, display text in high intensity, redefine keys, and give you control over the appearance of your display.

DEVICE=VDISK.SYS (RAMDRIVE.SYS for MS-DOS systems)
If you have lots of memory to spare, this function allows some of it to be used to emulate a very fast disk. The disadvantage is that you must remember to save the data, because it vanishes when the power goes off. Later versions of DOS (3.0) allow it be installed in expanded (XT usable) or extended (AT, 386) memory.

DEVICE=VDISK.SYS [size] [sec size] {max d e] [/E[:m]]
where size is the amount of memory to be used as a RAM disk (Remember to leave enough memory to allow your applications to run.) The other important parameter (if you have extended memory) is /E, which sets up the RAM drive in extended memory. The other parameters are sector size, maximum number of directory entries and m, the maximum number of sectors.

EXAMPLES:
DEVICE=VDISK.SYS
gives defaults size=64K, sector size 128K, and 64 directory entries in main memory.
DEVICE=VDISK.SYS 1024 256 512 /E
gives a large virtual disk in extended memory.
DEVICE=DRIVER.SYS
allows for external disk drives.
DEVICE=PRINTER.SYS
supports 'code page switching' (national language support, including French-Canadian), assuming that the hardware supports this feature. You must have at least EGA, but not all printers or software support this feature (DOS 4.0).

OTHER FUNCTIONS
BREAK=[on/off]
Default is off, i.e. DOS checks for Ctrl-Break only when an I/O function call is made. If set to ON, this check is made more frequently.

FCBS=X,Y
In DOS 1.xx, this was the way to open files. Some older programs still use it.

(*) LASTDRIVE=N
Sets the maximum number of drives you may access. The default is 'e' (=e). If you ever use the DOS command SUBST, you may need to use this function.

SHELL
Intended to load an alternate command processor, it is mainly used to increase the size of the environment, an area of memory where DOS stores information it needs to communicate with you and interpret the commands you type, including the path information, current prompt string and information which has been 'set' (perhaps through an application program). The default size is 128K, which can sometimes be too small. If you get the message 'Out Of Environment Space', use this function.
SHELL=COMMAND.COM /E:(size) /P (run autoexec.bat when command.com loaded).

Note that for DOS 3.1, size is given in multiples of 16 bytes, default 10 (160 bytes), maximum 62 (992 bytes). For later versions, size is in bytes. The minimum (and default) is 160, maximum is 32768 bytes. For example:

SHELL=COMMAND.COM /e:512 /p (DOS 3.2+)
SHELL=COMMAND.COM /e:50 /p DOS 3.1)

My current CONFIG.SYS (on an Ogivar 386 with 1MB of extended memory and running MS-DOS 4.01):

```
BREAK=ON
BUFFERS=35,8
FCBS=20,8
FILES=40
LASTDRIVE=z
DEVICE=RAMDRIVE.SYS 1000 512
  128 /E
SHELL=C:\DOS\COMMAND.COM /P
  /E:256
DEVICE=C:\DOS\ANSI.SYS /X
INSTALL=C:\DOS\FASTOPEN.EXE
C:=(150,150)
```

* * *

THERE'S LIFE IN THE OLD XT YET

by Robert Parkinson

(Continued from April)

ROM BIOS UPGRADE

After searching almost every Ottawa computer store in vain, I turned to mail order again. After a number of negative replies, I hit it lucky. A firm called Computerwerk Inc. handles a BIOS called the Award 8088/86 Modular BIOS Version 3.1C.

If you write them, they will send you a catalogue and an instruction sheet which explains how to determine the BIOS chip configuration you need. The BIOS is a bit expensive, so if you can find one locally, buy it.

This BIOS supports the enhanced keyboard and the 1.44M floppy drive. Judging by the ASCII strings embedded in the BIOS, it is a combination of the original Award XT BIOS and some AT BIOS code additions. It is larger than the original XT BIOS and will not fit in one 2764 EPROM chip. However, the firm will supply it in the configuration to match your particular motherboard. In my case, it came on two chips, the master chip to replace the original 2764 EPROM chip, and a 27256 EPROM chip to go into the ROM BASIC socket.

It took five minutes to install the chips. I crossed my fingers and booted the system. Success! I disabled the on-board BIOS on my floppy drive controller and let the new motherboard BIOS take over. There was now no need for any DRIVER.SYS or DRIVPARM entries in my CONFIG.SYS.

All programs, with one exception, recognize Drive B as a 1.44M drive. The one exception, strangely enough, is the FORMAT-F program from Mace Utilities, which still refuses to see Drive B as anything other than a 360K drive. DOS, Norton, PC Tools and MAXI Disk will all format Drive B correctly, either in 720K or 1.44M. System analyzers, such as SYSID or Quarterdeck's MANIFEST, fully recognize the drives.

What about the keyboard? Well, programs that check the BIOS now see support for the enhanced keyboard and work perfectly. The exception is DOS itself, and its ANSI.SYS. DOS does not check the BIOS area, assuming that an XT will not have an enhanced keyboard. The motherboard keyboard controller chip is the same original XT version and when IBM introduced the enhanced keyboard, it started only with later models of the AT and chose different interrupt 16h functions for the new extended keyboard codes.

My previous KBDR.COM would not work with the new BIOS. What now? The solution turned out to be a small TSR called NEWKEYS.COM that PC Magazine published for use on AT computers. With my original BIOS, this utility hung my system, but it worked perfectly with the new BIOS. So all is now well in the keyboard department.

MEMORY UPGRADE

As all XT users know, 640K of RAM doesn't go very far with a few TSRs and today's larger programs. The solution for the XT is "expanded" memory. But "caveat emptor" should be your motto and caution your guide, because not all expanded memory is created equally. In my case, I was given a good deal on an old Intel AboveBoard/PC with 2M of memory.

I logged on to the Intel bulletin board in California and downloaded their latest EMS 4.0 driver software and I was in business. Well, sort of.

The problem, which I knew when I bought the board, is that the new EMS or LIM 4.0 expanded memory management can be enabled in software and hardware (mine isn't) or software alone on an older EMS 3.0 board. If you have a choice, take the more expensive new hardware/software implementation, as it opens many new doors for you.

EMS 4.0 was introduced less than two years ago, so there are many boards still available that use only upgraded software drivers for old EMS 3.0 hardware. They are still advertised (falsely, in my view) as being fully EMS 4.0 compatible.

I use my expanded memory for a RAM drive (Drive D), using NJRAMDXP.SYS, a shareware program available on The PUB. It takes only 720 bytes of conventional RAM. The "XP" in the program's name indicates the enhanced processor version, designed to work with 286 and V20 processors. It is faster and uses less low memory than the standard NJRAMD.SYS, perhaps another good reason to upgrade to the NEC V20 processor!

I use the PC-Kwik Power Pak disk caching utility, which increased my disk performance index (using Core Test) by a factor of three and my data throughput by a factor of five. Of course, I still have the average access time of no better than 50ms. I have limited my disk cache to 512K and my RAM disk to about the same, so that almost 1M of expanded memory remains available for other use.

It is this "other use" that sharply differentiates between those expanded memory boards that implement EMS 4.0 in hardware and those that don't. Ideally, I would use Quarterdeck's new QRAM program to load my buffers (mouse, software, TSRs, device drivers, ANSI.SYS, etc.) into expanded memory, leaving me with almost 640K of free low memory free for programs, but QRAM and products like it will only do this if EMS 4.0 is implemented in hardware. The reasons are too complex to go into here, but true nevertheless.

HARD DRIVE CONTROLLERS

Although not strictly related to upgrading, let me add a few words about RLL hard drive controllers. The Western Digital controllers and most others are capable of two modes of RLL operation, "translating" and "non-translating".

Without getting too complicated, the translating version is designed to increase the number of sectors per track, thus speeding up data throughput. This is done in such a way as to fool DOS. The trouble is that, while fooling DOS, it may also fool some very useful programs, which can have disastrous results. Gibson's SPINRITE, for example, will not work with an RLL controller in the translating mode. Similarly, the shareware IAU19 program will trash your hard disk if you use it with a translating controller.

Normally, the controller comes set to translation as the factory default. I would recommend that you do as I decided to do: reset the controller jumper to disable translation. This may cut down your data transfer rate slightly, but if you are using a good disk caching program you won't notice it. You will have to do a complete reformat of the drive, starting with a low-level format, but this will give you a good excuse to try JOHNBACK, created by John Whelan. It's available on The PUB.

(Continued on Page 6)

THERE'S LIFE IN THE OLD XT YET

(Continued from Page 5)

If you have a drive greater than 32M and want to have only one partition, I would still recommend that you disable translation and upgrade to MS-DOS V4.01, which will handle disk partitions over 32M. As an alternative, you may want to try Compaq DOS V3.31, if you can find it, as it also handles partitions over 32M.

Another alternative, if you feel in a dangerously experimental mood, would be to try one of the commercial utilities that increases the hard disk cluster size, either by increasing the sector size above 512 bytes and/or the cluster size above four sectors. In this event, you can keep MS-DOS V3.30 (after all DOS 4.01 has a few bugs and takes a lot more RAM) and still get hard disk partitions greater than 32M. But, be careful! There is no guaranteeing how your valuable utilities, like Norton and SpinRite, will react.

FUTURE UPGRADES

Am I finished trying to "make a silk purse out of a sow's ear"? While the thought of a 386 machine with 8M of motherboard RAM running at 16 or 25MHz is awfully appealing, I'm enjoying myself seeing what I can do with an XT machine.

What else can I do now? I debated about an 8087 coprocessor chip; however, I don't do enough intensive numerical calculation work (CAD or spreadsheet) to make it really very useful. At this point, I am undecided about my next upgrade. I am certainly considering a tape drive for easy backups. I find the Colorado Jumbo unit most attractive, I can run it directly from my existing floppy controller, and it is available for a very reasonable price.

On the other hand, I would also like an expanded memory board that supports EMS 4.0 hardware. I could then use QRAM, or something like it, to free up as much of my low conventional memory as possible. This should eliminate any "Program Too Big" messages.

I also need more hard disk space (who doesn't?) and may also add a second hard drive. My existing controller will handle it with ease. I would, however, ensure that the new drive was a high-performance model that I could use later in a 386 system.

HINDSIGHT

Hindsight, always being 20\20, I would do things a little differently if I was to start out now to put together an XT system, assuming that an XT was the best way to go (a very dubious assumption).

I would certainly take a close look at the Everex XT turbo motherboard. It uses a NEC V20, running at 12MHz, and is upgradable to 15MHz. It has a large American Megatrends (AMI) BIOS which supports the 3.5" 1.44M drive and the enhanced keyboard.

It comes with a built-in high-density floppy controller (which can be disabled if you want to add your own) for two drives and has 1M of 100 ns. RAM. I doubt, however, if this 100 ns. speed would suffice at 15MHz.The extra RAM feature is very nice, even though DOS will use only 640K, because QRAM and other such programs (some shareware) will use the high memory, not just to load data, but also programs such as device drivers, TSRs, etc.

This extra 380K (depending on how you calculate it), coupled with one "real" EMS 4.0 expanded memory card, would be ample for any normal DOS usage, where we are not talking about OS/2, multitasking, and so on. I haven't used the Everex board, and there well may be others like it, but the attractive features make it worth careful investigation.

ADDRESSES

ASSOCIATE COMPUTER SUPPLY COMPANY INC.
3644 Tibbett Avenue
Riverdale, NY
USA 10463
(212) 543-3364

KOMPUTERWERK, INC.
851 Parkview Blvd.
Pittsburgh, PA
USA 15215
(412) 782-0384

NORTHGATE COMPUTER SYSTEMS, INC.
P.O. Box 41000
Minneapolis, MN
USA 55441
(800) 338-8383

* * *

WORDPERFECT 5.0 SOLUTIONS

by Bonnie Carter

TIPS FROM OPCUG MEMBERS

From Thomas Greene

In List Files, after you have marked a number of files with an asterisk, you can move from one tagged file to the other by pressing the TAB or SHIFT-TAB keys, depending on the direction you want move in. This is handy if you are working with a large directory where not all the tagged files are displayed, or if you have done a word search and you would like to find out which files have been tagged.

If you want to see only specified groups of files, select 7 (New Directory) and press ENTER until you see the line at the bottom left of the screen display the familiar *.*. At this point press the END key and then backspace to edit the line. If you want to see all files that end in TXT, backspace once and type TXT and press ENTER. Only TXT files will be displayed.

Suppose you want to search through a long document for a specific phrase. Move the cursor down through the document a couple of lines. Press the Search key <F2> and type the word or phrase you want t locate followed by ENTER. The curso moves to the word or phrase. Keep pressing the F2 key to continue your search.

IF you want to make sure that the numbers in the table of contents match the page numbers they are supposed to, move to the table of contents and choose the page number you want to check. Press the GOTO key CTRL+HOME and type in this number and then ENTER. Check the page number on the status line to see if it matches the one in the table of contents. Press the GOTO key twice and you will be returned to the Table of contents. Choose the second number and repeat the above process.

From John Whelan

If you wish to delete or move all the files in a particular directory, simply press <ALT F5>. An asterisk will appear beside all the files in the directory. Then it's a simple matter of choosing 2 (Delete) or 3 (Rename\Move). You will be asked if you want to continue with the deletion or the move. Answer (Y)es or (N)o, depending on what you really set out to do.

* * *

BACK TO BASICS

By Harry Gross

BASIC OUTPUT STATEMENTS

The outputs to be discussed here are the PRINT and LPRINT statements and their variations PRINT USING and LPRINT USING. The discussion here will use PRINT for illustration, but LPRINT will behave the same, except:

a) You should have a printer.

b) Column widths other than 80 characters are possible.

Consider the following small program:

```
10 CLS                    ' clear the screen
20 a=0.5: b=-0.5: c=3.55: A$="abcd": B$="1234"
30 PRINT a b c A$ B$
40 PRINT a,b,c,A$,B$
```

Using GW-BASIC or IBM BASIC, run it, and the screen will display:

```
 .5 -.5 3.5 abcd1234
 .5   -.5   3.55          abcd    1234
```

On the other hand, using Turbo BASIC we get:

```
 .5 -.5 3.54999992316284 abcd1234
 .5 -.5      3.54999992316284    abcd    1234
```

The difference in the spacing between the lines arises from the presence of the commas. Without them, numbers are written with one space between them, and strings are packed. With them, the program will start each value at the next available default tab, which are spaced 14 columns apart, at columns 1, 15, 29, 43, 57, 71, and so on. For the screen, the program wrapped the last item "1234" around to the next line because we exceeded the 80-column capability of a standard monitor.

We also see that something strange happened to our value of 3.55. The program insists that it is 3.54999992316284. For a computer, that is good enough, but it does mess up our nice neat columns. So, for better control of our output, we may use PRINT USING.Add these lines to our program above:

```
40 PRINT USING "##.#  ##.#  ##.##  \ \ \ \";a,b,c,A$,B$
```

and the output in either version of BASIC will appear as:

```
 .5  -.5   3.55   abcd   1234
```

The string inside the quotation marks is referred to as an "image string" and gives us better control of the output format.

We have the following image descriptions available to us:

```
####        - for formatting integers.
###.##      - for decimal numbers. The decimal marker will set
              the number of significant figures to be printed.
```

One column for each # and decimal point will be used.

```
+###.##     - Include the sign in the listing.
##.###      - Use a trailing minus sign, if negative.
              (Accountants like this.)
###,.##     - Group numbers by threes. i.e. 1,100.34
$$###.##    - Dollar sign, butted up against the number.
###.###^^^^  - 4 carats, use scientific notation.
```

And for string output:

```
\  \        - Use as many columns as between and including
              the back slashes, say a total of n columns. If the
              string is longer than n characters, then use the first
              n  characters.
&           - Use as many columns as needed.
```

If a number is printed with a leading &, say %123.45, then the number is too large for the allocated space.

Try playing with various combinations of the above to see what their effect is. Any combination is possible, and leaving blank spaces in the image string will space out the columns as needed.

The image string need not be in the statement. We can define it as a variable and then use the same image in several places.

```
10 Image$[1]="   ####.###   $$######,.##   \    \"
20 Image$[2]=Image$[1] + " $ "
30 PRINT USING Image$[1];  Volume,Price,Date
.............
140 PRINT USING Image$[1}; Volume,Price,Date,  Client
```

One shortcoming is the inability to right or centre justify a string in a field. So, say we have a field 10 spaces long, and a string B$, of any length up to 10 spaces.

```
10 B$="xxxxx"
20 B$=SPACE$(10)+B$
30 B$=right$(b$,10)
40 print using "\         \";b$
```

and our output is now right justified. This can be combined into one function:

```
10 DEF FNR-Ju(n,A$)=RIGHT$(SPACE$(n)+A$,n)
40 PRINT USING "\        \"; FNR_Ju(10,B$)
```

For centre justification:

```
10 B$="xxxxx"
20 bl=LEN(B$)
15 bs=FIX((10-bl)/2)
17 bf=bl=bs
20 B$=SPACE$(bs)+BS+SPACE$(bf)
40 PRINT USING "\        \";B$
```

The fiddling is to allow for strings and spaces of an odd length, and as a one-liner:

```
10 DEF FNC_Ju(n,1,A$)=
  SPACE$((FIX(n-1)/2))+A$+SPACE$((FIX(n-1)/2))
100 B$="xxx"
110 Image$"\"=SPACES$(8)+"\"
120 PRINT USING "\        \"; fnc_jU(10,len(b$),b$)
```

We can even define our images as functions.

```
10 DEF FNImage$(n)="\"+SPACE$(n-2)="\"
20 Im$=FNImage$(8)+"  "+FNImage$(6)+"  "+FNImage$ (10)
30 PRINT USING
  Im$,A$,FNR_Ju$(6,B$),FNC_Ju$(10,LEN(C$),C$)
```

The main reason for using this type of formatting, as round about as it may seem, is to avoid counting spaces when setting up an image.

One last useful note. If a print statement is terminated with a semicolon (;), the no carriage return and line feed are sent out, and the next print statement will start in the next column on the same line.

So:

```
10  PRINT "Total - " A$ " ";
    ...
    ...
100  PRINT USING "$######,.##"
```

will produce:

```
Total - Widgets   $1,235.98
```

There are many variations on these functions. Practice using them and refer to your manual for more details. In the next column, we will start on loops, really the heart of most programming.

* * *

OTTAWA PC USERS' GROUP

EXECUTIVE		
President	David Terroux	238-4895
Past President	Stu Moxley	592-4933
Treasurer	Tony Frith	671-0401 Weekends
Secretary	Claude Jarry	521-3366
Facilities Officer	Douglas Poulter	745-8768
Membership Chairman	Carl-Henri Gomez	731-1462
Publicity	André Cyr	561-5207
Software Librarian	Chris Taylor	723-1329
Newsletter Editor	Bonnie Carter	236-1015
BBS System Operator	Mike Schupan	820-0293
BULK PURCHASING	Terry Mahoney	225-2630 226-2615 FAX
ASSISTANTS		
Software Assistant	John Ings	235-8132
Newsletter Assistants	Marc Riou	733-2092
Publishing Consultant	Ryszard Cimek	232-3057
Contributing Newsletter Editors	Chris Taylor	723-1329
	Ann Falkner	722-3226
	David Smith	837-9291
BEGINNERS' SESSIONS	Eric Clyde	749-2387

DISK OF THE MONTH

One-year subscription	(10 DOMs) 5 1/4"	$25.00
One-year subscription	(10 DOMs) 3 1/2"	$35.00
Individual 3 1/2" disk surcharge		$2.00

THE OTTAWA PC USERS' GROUP MEMBERSHIP APPLICATION - Please Print

Last Name: _____ First Name: _____

Mailing Address: _____

Postal Code: _____ Telephone - Home: _____ Office: _____

Profession: _____ Date of Birth: _____ Sex: M _____ F _____

Disk of the Month: YES _____ NO _____ Size: 5 1/4" _____ 3 1/2" _____ Amount Enclosed $ _____

Are you: A new member? _____ Renewing your membership? _____ Sponsor's Name: _____

What in particular interests you in the Group? _____

Can you help in Group activities? Check off the activities that apply: Programming Language Instruction _____

Newsletter Input _____ Memberships _____ Software Library _____ Promotion/Publicity _____

Hardware Techniques _____ Meeting Locations _____ Agendas & Speakers _____ Advertising _____

Bulletin Board _____ Other _____

What hardware/software do you own and/or use? _____