

# PolyLetter



PolyLetter 90/1

Page 1

JAN/FEB 1990

## *Editorial*

It was gratifying to receive some feedback on my editorial on recycling. [See Russ Nobbs's letter.] Who'd a thunk I'd get support for non-computer ramblings? Well, with that kind of encouragement, here's more food for thought.

### *An Open Letter to Quaker Oats*

Quaker Oats Company  
Post Office Box 9003  
Chicago, IL 60604-9003

Greetings,

I have been a long standing loyal consumer of your Old Fashioned Quaker Oats. I was proud to announce to all my friends that your product had a 100% recyclable container.

Unfortunately, I can no longer make that claim. With the introduction of plastic rims on your boxes, you are moving in a direction OPPOSITE to what is needed for preserving our environment. You force me to switch to bulk brands. If you restore the original packaging design with a string to open the boxes in the near future, let me know and I'll return to buying your oats. But, don't wait too long. I may get comfortable with a new brand, one with environmental sensitivity, and give them deserved loyalty. -- Sincerely, Ralph E. Kenyon, Jr.

Speaking of the environment, how about that Prop 65 California just passed? As I understand it, companies are required to provide "reasonable notice" to the general populace if they produce a product containing hazardous chemicals or if they discharge hazardous chemicals into the environment. Perhaps one of our California readers can say more about it. I heard on ABC News that the makers of Liquid Paper changed their product formula rather than

announce that they had a hazardous chemical in it! Sounds like Prop 65 might benefit all of us.

## *Announcements*

Bob Bybee has just announced an upgrade to the PolyEmulator. For EGA and VGA users, PM.EXE version 1.3 now supports true PolyGraphics using a special character font file. The user can program custom character sets as well. The upgrade fee to prior owners of Version 1.1 or 1.2 is \$20.00. Contact Poly Peripherals, 5011 Brougham Court, Stone Mountain, GA 30087, Phone 401-498-3556.

## *Letters*

PolyLetter

February 15, 1990

Ralph,

I meant to tell you that it was a good editorial in 89/4 on recycling. Finding ways to recycle as much as possible is an important job we all need to do. There is no "away" when we "throw things away".

As a small business doing a lot of mail order shipping, we actively solicit used packing material from neighboring businesses. We have one recycler supplying us with packing material they collect and use plain old newspaper we bring from home when shredded packing is not available.

A computer friend was surprised by our bins of aluminum and glass next to the office refrigerator. He said he guessed "it wasn't worthwhile for big business to bother with that kind of recycling". I said I thought the big businesses needed to be the first to do it to set the standards for industry.

I'm often in public buildings for conventions, or even parks and school campuses and see aluminum cans and glass

bottles in the trash or gutter. Where are the "glass only" and "Aluminum only" bins in these public places?

I recently subscribed to the new "Garbage Magazine" which begins to show some solutions to these problems.

As you may remember, I adopted WordStar on my CP/M machines and have stayed with it on my DOS machines. As WordStar has added features, I've been pleased to see that they have finally gotten around to adding some of the early Poly Edit features. I now have ESC <chr> shorthand that works very much like the Poly. There is also a ^O that does flush right like FORMAT's {jr} justify right command. The undelete feature (^U) only marked blocks that have been undeleted so I still miss being able to undelete one or two characters "overruns" when I hold down the back arrow key. Oh well, some of the other features kind of make up for it. And, really, Ventura is a lot more powerful than format ever was.... Keep up the good work. -- Russ Nobbs, Spokane, WA.

### **Benchmarks**

by Bob Bybee

I must thank you for the extensive coverage given to PM, my PolyMorphic Emulator program, in the last issue of PolyLetter. PM has been a "best seller" for me, thanks in part to your support. I have received good feedback from my customers, and PM seems to be doing well in the field.

When I began the PM project, I wasn't really expecting the thing to run at high speed. I expected that to run close to the speed of a "real" Poly, I would need to use a hardware solution, such as a V20 chip or a plug-in Z80 board. PM surpassed even my expectations when I ran some benchmarks recently, and discovered that a fast 386-based PC can actually run PM faster than a Poly! The software-only solution, PM, has succeeded.

My benchmarks were two BASIC programs: one that added up all numbers between 1 and 1000, and one that performed a "prime number sieve" of 1000 integers. I admit these are simple benchmarks, but they do represent compute-bound programs with no disk activity. They should provide a good index of BASIC program performance, and at

least a rough measure of other CPU-bound programs.

Here are my results, with a "real" Poly equalling 1.0:

		<u>relative CPU performance</u>	
V20:	0.10	NEC V20,	4.77 MHz
286:	0.57	80286,	10 MHz
386/20:	0.91	80386,	20 MHz
Poly:	1.00	a "real" Poly	8813
386/25:	1.06	80386,	25 MHz
386/20C:	1.10	80386,	20 MHz w/cache

Note: a "real" 4.77 MHz IBM-PC using an 8088 should be somewhat slower than the V20 tested. A "turbo" PC or XT clone that can run at 8 to 10 MHz should fall somewhere between the V20 and the 286 speeds shown.

Now, how does PM's disk speed compare with a Poly's? A PC's disks are generally much faster:

- A PC floppy is about 1.8 times as fast as a Poly SSSD 5" floppy.

- A PC hard drive is 3 to 6 times as fast as a Poly hard drive.

A true system performance measurement must include both disk I/O speed and CPU speed, weighted according to how much of each a "typical" program uses. One test which does this in the PC-clone world is the Norton SI (System Information) program. SI compares the CPU speed of a system to that of a 4.77 MHz IBM XT, and calculates the Computing Index (CI) for that system. (A system with a CI of 2.0 would run twice as fast as the proverbial XT.) Likewise, it measures the Disk Index (DI) of a system compared to an XT, then calculates an overall Performance Index (PI) using the formula

$$PI = (0.66667 * CI) + (0.33333 * DI)$$

This weights the overall performance, with CPU speed being twice as important as disk speed. No benchmark can really be said to be "typical" for all purposes, but SI is widely accepted for comparing PC clones.

Using that formula for PI, and assuming we're comparing a fast PC hard disk to a Poly hard disk (so that PM's Disk Index is 6.0), here is how PM measures up on various PCs:



overall performance

Poly: 1.00  
 V20: 2.07  
 286: 2.37  
 386/20: 2.61  
 386/25: 2.71  
 386/20C: 2.73

Of course, these figures pretend that a fast disk can make up for a slow CPU. Depending on the program, this may be true in some cases.

I have not yet had the opportunity to run PM on an 80486-based machine. But I've heard that such systems run 2 to 4 times the speed of an 80386. Given that, PM might potentially run at a Performance Index of 4.67 times as fast as a Poly! Anyone who has access to a '486 and would like to try this, give me a call.

There's an old saying in this business: "Benchmarks don't lie, but liars write benchmarks." I realize that the only way to get a true measurement of how your programs will run on a given system is to actually run them. But I hope this gives your readers a feel for what PM can do in the real world. Most people using a PC for serious business work today would choose at least a fast '286, and preferably a '386-based system. PM can give good performance in any of these environments.

Thanks again for publicizing PM, and continued best wishes for PolyLetter.

***Evaluating Benchmarks***

When I first read Bob's benchmark claims, my only experience was with timing my clone at both 4.77 and 8 MHz. I am running a NEC V-20 cpu on my clone and when the turbo mode is turned off, it runs at 4.77 MHz, just like an ordinary PC. My earlier timing comparisons produced a difference of about 1:16 in timing, but using a different benchmark program. Since I had gotten a 1:16 timing ratio, Bob's claims didn't seem quite to agree with my experience. 1:16 is .0625; Bob had claimed .10 as his emulator cpu performance factor. There is a 40% difference. .0625 doesn't round up to .10; it rounds up to .07. So, where did Bob get his .10? I ask.

Well, I decide to run one of Bob's benchmark programs to see if the result was very different from what I had experienced.

He said he added up the numbers from 1 to 1000, so I tried the following program.

```
10 FILE:2,LIST
20 PRINT "PRESS ANY KEY WHEN READY", \Z=IMP(1) \PRINT
30 Z=TIME(0)
40 FOR I=1 TO 1000 \X=X+1 \NEXT
50 PRINT TIME(0)/60
60 PRINT:2,CHR$(?), \PAUSE 30 \GOTO 60
```

I ran the program with the TURBO mode OFF and by using the sweep second hand on my watch for the timing. I also ran the programs with timing points which read the real-time clock and reported "more accurate" times. My sweep second hand results agreed to within a second of the internal timing measurements, so I present the more precise figures as possibly being more accurate. I also ran the program on a Zenith 386/25 machine. Here are the results.

Poly	Emulator 4.77	8.0 MHz	25 MHz 386
8.41 Seconds	125.5 Seconds	63.11 Seconds	5.88 Seconds

Dividing the Poly's performance by each of these by the gives the computing index (CI).

Poly	Emulator 4.77	8.0 MHz	25 MHz 386
1.00	.067	.133	1.43

The above results should be pretty good, since no Disk I/O is required during the timing portion of the program.

To get some results for the disk performance I ran the Poly Sniff command, which just reads the disk, one sector at a time, looking for errors. Since this operation does almost nothing except read the disk, it should give a fairly close result (provided no errors are detected). Exec(AIS) has the capability to specify the starting sector on the Sniff command. For this test on the floppies, I executed the following command.

```
Sniff 1 150
```

This instructed the system to read 150H (336) sectors, one sector at a time. On the hard disk, I upped the amount to 900H sectors, but divided the result by 6.857 to obtain a comparison for the floppy figure. Dividing each of these into the Poly's Floppy disk time gives the Disk Index relative to a Poly 8813 SD machine. Here are the results.

	Poly	EM 4.77	EM 8.0	
Floppy	63	58	33	Sniff 150H Sectors
Hard disk	27	155	110	Sniff 900H Sectors
Hard disk	3.94	22.6	16.04	Sniff 150H Sectors

## Relative Disk Performance Index (DI)

Poly FD	Poly HD	EM 4.77 FD	EM 4.77 HD	EM 8.0 FD	EM 8.0 HD
1.00	16.0	1.09	2.79	1.91	3.93

Look at the Poly Hard disk performance factor! Wow! No wonder I'm spoiled!

Note: I got the same time, (63 Sec.) on both SA-400 floppies, and 96tpi floppies. The performance limit seems to be in the rotation rate and the data transfer rate for the single density controller. The Rotation rate is the same for PC drives, but the data transfer rate is doubled. Hence, we should expect to see about 1/2 the Poly's time as a limiting factor. The step rate for the pc drives is 6ms while the step rate of the Poly drives is 40ms; the Sniff test does not exercise this difference very well. We can't assume that a faster step time will dominate the disk performance. I couldn't think of a way to exercise this without involving a lot of computing time as well, so I kept it simple.

I also used the Sniff test on 720K 3.5 inch floppy drives on both my clone and the Zenith. The result was remarkable.

Sniff test - 150H sectors 720K floppy

Clone	Zenith
4.77	8.0
60	33
	25MHz 386
	33

Bob gave us a formula for combining computing index and disk index to obtain an overall performance index.

$$PI = (0.66667 * CI) + (0.33333 * DI)$$

Here are the combined results using the figures I obtained.

Configuration	2/3 * CI	+ 1/3 * DI	= PI
1.84 Poly FD	2/3 * 1.00	+ 1/3 * 1.00	= 1.00
1.84 Poly HD	2/3 * 1.00	+ 1/3 * 16.0	= 6.00
4.77 V-20 FD	2/3 * .067	+ 1/3 * 1.09	= .408
4.77 V-20 HD	2/3 * .067	+ 1/3 * 2.79	= .975
8.00 V-20 FD	2/3 * .133	+ 1/3 * 1.91	= .725
8.00 V-20 HD	2/3 * .133	+ 1/3 * 3.93	= 1.40
25MHz 386 FD	2/3 * 1.43	+ 1/3 * 1.91	= 1.59

Let's compare these to the figures Bob

reported.

## Relative CPU performance (CI)

Bob's results		PolyLetter
V20	0.10	NEC V20, 4.77 MHz .067
V20	----	NEC V20, 8.00 MHz .133
286	0.57	80286, 10 MHz ----
386/20	0.91	80386, 20 MHz ----
Poly	1.00	a "real" Poly 8813 ----
386/25	1.06	80386, 25 MHz 1.43
386/20C	1.10	80386, 20 MHz w/cache ----

## overall performance index (PI)

Bob's		PolyLetter
Poly	1.00	(1.84 MHz) 1.00 / 6.00
V20	2.07	(4.77 MHz) .408 / .975
V20	----	(8.00 MHz) .725 / 1.40
286	2.37	-----
386/20	2.61	-----
386/25	2.71	(25MHz) 1.59 / ----
386/20C	2.73	-----

Well, to quote Bob's Quote, "Benchmarks don't lie, but liars write benchmarks." So, there you have the results of two different liars. We agree on one thing -- Poly's run at 1.00. Who ya gonna believe? Field results are probably the best index of performance, so send in any actual test results you get. But, describe the situation in detail so we can see exactly what the observed results measure.

**PM BUG**

While fooling around with the Emulator and PMU going to DOS, I returned to Exec only to get the familiar (Error 0306). Wow! How'd that happen? No disk or door open! On the hard disk no less! Well, I rebooted and tried again. But, I realized what I had done. When I had gone to DOS using PMU, I had changed the default directory to something other than the one where PM.EXE and SYSTEM.PM live. The result, is that when I went back to PMU with the EXIT command from DOS, and then went back from PMU to Exec, the emulator couldn't find its system disk file. The best way to avoid this "bug" is to restore the PM.EXE default directory before returning to the emulator.

I also found a way to get back in without rebooting. I hit CTRL-Z and brought up the front panel. Then I set the program counter to USER with the command SPJ3200. Then a G for "GO" put me back in PMU. The DOS command got me back to DOS,



and I set the path back to its desired value. Of course, you must be running in ENABLEd mode to do this.

Also, some people might argue that this "bug" is just an unimplemented desirable feature. Suffice it to say, knowing one way to get into trouble, and how to recover from it adds to our ability to use the system effectively.

### ***Printer Sharing***

I've had a pc-clone for some time now, but haven't done very much with it. I've transferred files back and forth to the Poly, and have converted some Poly BASIC programs to run under DOS; but, that's about all. I have recently begun swapping text files with associates, and have begun to use Word Perfect on the clone; but until just lately, I never bothered to print anything directly from the clone. I would send the file to the Poly first; then I would use the Poly to print the file. Now that I have the Poly Emulator running on my PC-clone, I will want to print directly from the clone to the printer. But, I want to continue to send files back and forth from the Poly to the clone. Here's how.

Let's see, the requirements are to be able to send files back and forth between the Poly and the clone; to be able to print directly with the Poly; and to be able to print directly with the clone. But, I don't want to have to plug and unplug different cables all the time. I won't mind having a couple of switches to flip, but I don't want to have to change any wiring.

I had an "A-B-C-D" switch connected to the Poly. One terminal went to my label printer. -- I have two printers connected to the Poly, one on each of two printer mini-cards. One is my main document printer, a very nice DataProducts 8050 color printer, which has a black ribbon in it and is loaded with standard 8-1/2 by 11 paper. The other is a Prism-80 and is loaded with continuous feed labels. I can switch between them with the Printer command, and have command files and Gnomus macros written for this purpose. -- Anyway, the "A-B-C-D" switch was connected to one of the Poly's printer mini-cards; it was also connected to the label printer, the clone, and a test plug.

Because the Poly's header was crossed, the cable I made up to go between the clone and the Poly just had straight wires. The Poly did all the crossing. Also, all the printer cables were straight through too. This meant that these cables could not be used to print from the clone. The cables needed to be crossed. But, I needed straight cables to continue to talk to the Poly.

The solution I hit upon seemed pretty neat. I needed to have a switch which would connect the Poly to the printer, the clone to the printer, or the Poly to the clone. The wires from the Poly to the printer needed to be straight through, as did the ones between the Poly and the clone. But, the wires between the clone and the printer needed to be crossed.

At first I wanted to do this with one switch. It would have to be a "Y" switch with 3 positions -- A-B, A-C, or B-C. Well, I haven't been able to find one like that. I hit upon using two switches, an "X" switch and an "A-B" switch. I got an "X" switch from Al Levy and set it up so that the "X" switch would be connected such that in one position it had the Poly connected to the clone, and in the other position it had the Poly connected to a printer AND the clone also connected to a printer. I put the "A-B" switch connecting the one printer to the two "X" switch outputs. When the "X" switch had the printer cables connected to both computers, the "A-B" switch let me select which computer got to use the printer.

Now, the cable going from the poly to the "X" switch is straight, as is the one going from the clone to the "X" switch. So, when the "X" switch has the Poly and the clone connected, they can talk to each other. Also, the cable going from the "A-B" switch to the printer is the original straight one I used between the Poly and the printer. Because the cable going from the clone to the printer must be crossed, the present configuration leaves only one place to put it. It must go between the "X" switch and the "A-B" switch. While I was at it, I made a general cable with all the pairs crossed, not just the ones the printer would need. The end result is what is called a 'null-modem' cable. I can disconnect this cable and connect it to another pc and have a null-modem.

By inserting the null-modem cable between the "X" switch and the "A-B" switch, I can use it to make the proper cross-overs needed between the clone and a printer. Now, when I set the "X" switch to connect the computers to printers, one "A-B" switch position goes straight through to the Poly, or crossed through to the clone. But, when the "X" switch is set to connect the computers, it is straight through. Two switches let me put my printer on line with either computer, or let me connect the computers to each other.

### *Primitive Displays*

Well, once again, it's PC bashing time! Bob Bybee announced an upgrade to the Poly Emulator (Version 1.3) for EGA and VGA users which gives one "true" Poly graphics when running the emulator. Of course I just had to have the upgrade right away.

When I first looked at Bob's graphics implementation, my first reaction was "This just doesn't measure up to the Poly!"; the graphics characters aren't similar enough at all. They aren't sharp enough, and don't have all the definition of Poly graphics. So, I took a look at his font file and compared them to my printer graphics files I had created for printing Poly graphics characters on the printer. They just didn't match. Bob's characters had double wide dots and a corresponding loss of definition. The Poly's "!" just doesn't have a "fat" stick! I'm not faulting Bob for just adopting IBM's character font and filling it in with a couple of missing characters, his product is a reasonable implementation of Poly graphics, given the limits of the IBM font.

It turns out that the IBM EGA display uses an 8 by 14 dot matrix to display characters, while the VGA display uses an 8 by 16 dot matrix. The character font uses a two dot wide philosophy to display vertical lines or dots. Bob says that a character is too dim when only one dot is used for the width, but I disagree. Also, the 14 (or 16) dots of vertical resolution includes the blank lines between the characters.

Most EGA characters blank the top two rows and the bottom three rows of dots to provide the vertical space between characters. This leaves the characters displayed in an 8 wide by 9 high dot

matrix. But because a two dot wide philosophy is used, this gives an effective resolution of about 6.5 wide by 9 high. It wouldn't be only 4 wide because the two dots of a width could be placed at 7 positions, and two such vertical bars could only be placed in 6 positions. (I'm assuming only one dark dot is needed between bright pairs.) Averaging gives the 6.5 figure.

How does this compare to the Poly?

Years ago, a program was included in an early PolyLetter "disk of the month" which reproduced the Poly character matrix. The program, by John McGaw of Anchorage, Alaska, displayed the Poly character font exactly as it was in the Poly VTI Character Generator Roms. The Poly uses an 8 wide by 11 high dot matrix for characters and uses a single dot philosophy.

By comparing the character dot resolution by height and width, the Poly gets a relative character resolution factor of 1.225 over the IBM EGA display (as the current EGA font is implemented). If we compare areas, the Poly has a resolution factor of 1.5!

The Poly's resolution is actually better than EGA because of the number of dot rows between rows of characters. The Poly blanks 4 rows of dots, for a total character display matrix of 8 wide by 15 high dots. Poly's low resolution graphics uses 4 dots wide by 5 dots high for each pixel. EGA is only capable of 8 wide by 14 high; so, once again, a pc-clone is a step backward from the Poly! Of course, with its 80 by 25 display, the pc-clone can show 95% more characters than the Poly. Personally, I'd settle for fewer, but better looking characters any day.

I want to see the familiar crisp Poly font on my emulator as well as on my Poly. After all, we Poly users are just not used to settling for second best.

Well, I remembered John's program and promptly got it out. I modified it to send it's output to a file in a manner compatible with Bob's EGA font file. Then I replaced all the Poly characters in the EGA font file with the font generated by John's program. Bob had already removed the 15th row of dots from the middle two graphics characters, so these were ok as



is. Then, I copied this new font file back to the Clone and named it POLY.FNT. (I did all this edit work on the Poly itself.) To approximate the Poly's original display, I selected the color combination of B-WHITE on BLACK. Wow! Such an improvement! Now, it LOOKS like a Poly display with a bright, crisp high definition character font. There is a 25% loss of separation between rows of characters, but that's not bad considering we get the true Poly Character font.

Of course there is one weird special effect when one uses the DOS command in PMU to exercise a DOS program. The Poly character font is still connected, so DOS application programs, which use the IBM extended character set, get Poly graphics characters instead. A BOX around text in a pop-up program becomes a line of Poly graphics characters instead. It's a small price to pay for having the true Poly font on the Emulator.

If anyone wants the Poly.FNT file for their emulator 1.3 upgrade, send me \$5 and it's yours (Specify Poly or PC disk).

## HELP!

In this section I share with you the help system files I have built up over the last few years. (The entire system is included with Abstract Systems Exec.)

### HELP COMMAND TYPE

HELP file for system command "TYPE"

The "TYPE" command displays a file on the system screen. "TYPE" waits after each page with ":". At the ":" ESC exits.

Syntax: "TYPE [<n>path<file.TX>]" (RETURN) (see HELP PRINT)  
'n' is a drive number and file is a text file.

"TYPE [file]" displays a file on the system resident drive.  
"TYPE [<n>file]" displays a 'file' on drive 'n'.  
"TYPE [<n>path<file>]" displays a 'file' on drive 'n' and in subdirectory path.

Minimum size: "T" Example "T <2>LETTERS<Disk>"

### HELP COMMAND UNDELETE

HELP file for system command "UNDELETE"

The "UNDELETE" command undeletes all deleted files in a directory.

Syntax: "UNDELETE [<n>path<file.DX>]" (RETURN) (see ARISE)  
'n' is a drive number and file is a directory file.

"UNDELETE [n]" undeletes all deleted files on drive 'n'.  
"UNDELETE " undeletes files on the system resident drive.  
"UNDELETE [<n>path]" undeletes files on drive 'n' and in subdirectory path.

Minimum size: "UM" Example "UM <2>LETTERS.DX"

## For Beginners

The Poly WordMaster system presents options using a menu system. It is possible to use the Poly exclusively in a word processing mode just by selecting options from the menus. But, when you turn on the Poly and get the '\$' prompt, it would be nice to know what you can do.

But, knowing what you can do depends, in part, upon knowing what's there to do it to or with. Poly's (and other computers) store stuff in files on disks. The files are indexed in a directory. There are two main types of files. One kind of file is a program which can be "run". The other kind contains data or stored information. These can include text files, data used by programs, etc.

When the computer is first turned on it goes through a process called "booting up", in which the main operating system program takes control. The Poly's operating system resides on disk, so a "system disk" must be in drive # 1 for the Poly to complete the boot-up process. Once the Poly is booted-up, it then waits for you to tell it what to do. Each choice is started by entering a system command. What are the choices for a Poly?

The Poly has 38-42 choices, depending upon which version of the operating system you are using. Most commands are the same in all versions. I have been publishing detailed help files for various system commands, but for the present purposes we only need a brief description. I have arranged the system commands into 7 functional groupings.

### 1. Directory Commands.

These commands affect the contents or display of directory information. These commands affect or operate on entire directories, including the main directory of a disk. 'LIST', 'list', and 'DIRECTORY' list the names of files in a directory; 'DIRECTORY' sends the output to the printer. 'DLIST', which is not in early



operating systems, lists the deleted files in a directory. 'UNDELETE' recovers all deleted files in a directory. 'SetSys' marks all files in a directory as "system" files. 'UnSys', which is only in Exec/{A|S}, marks all files in a directory as "non-system" files. The '#' (wild card path) command sets the wild card to a drive or drive and subdirectory (# <2<SUB).

## 2. File Handling Commands.

These commands allow performing various operations with files. 'EDIT' allows creating or changing some files. 'TYPE' "types" the contents of a file on the system screen; 'PRINT' "prints" a copy of a file on the printer; 'COPY' allows making a copy of a file; 'RENAME' allows changing the name of a file; and 'DELETE' allows deleting a file.

## 3. Printer Control Commands.

These commands allow controlling the system printer. 'PAGE' instructs the printer to move to the top of the next page; 'Printer' allows connecting different printers, displaying or changing different printer parameters, etc.; and 'WRITE' (which is only in Exec/{A|S}) allows sending a comment from the keyboard direct to the printer.

## 4. Program Control Commands.

These commands permit controlling the operation of the program loaded into memory. Just typing the name of a program <NAME> tells the system to run that program. A running program can be interrupted with the <CTRL-Y> interrupt command. This stops the program from running (in most cases). The 'GET' command allows loading a program into memory without starting it. The 'START' command allows starting or restarting a program which is already loaded into memory. The 'REENTER' allows reentering a program which is already in memory. The 'CONTINUE' allows resuming a program which has been interrupted with the <CTRL-Y> command. The 'SAVE' command lets one "save" a program or data from memory to disk. The 'DUMP' command makes a printed copy of the program in memory using hexadecimal numbers. The 'ZAP' command "erases" the program in memory. And, the 'RESET' command resets the computer stack pointer to the beginning. The 'ReStart' command, which is only in Exec/{A|S}, allows resuming an aborted command file.

## 5. Keyboard Processing Commands.

These commands allow changing the way the keyboard processes characters. The 'fold' command tells the keyboard to "fold" lower case letters into upper case; this command remains in effect until cancelled. The 'flip' command tells the keyboard to "flip" the case of letters typed from lower to upper and from upper to lower; this command remains in effect until cancelled. The 'FULL' command cancels the effect of the 'fold' and 'flip' commands.

## 6. Disk Maintenance Commands.

These commands help with the maintenance and administration of diskettes. The 'INIT' command is used to initialize ("format" in IBM parlance.) a disk before data can be stored on it. The 'DNAME' command allows putting a name on a disk. 'PACK' recovers the space occupied by deleted files. Once a disk is PACKed, deleted files can no longer be recovered. The 'IMAGE' command allow making a copy of the entire data contents of a disk. And, the 'Sniff' command reads the used portion of a disk looking for errors.

## 7. System Functioning Commands.

These commands affect the way the system operates, or provides access to additional information about the system. The 'Auth' command controls security and access, allowing adding and changing users and user passwords. The 'boot' command allows rebooting or booting to another drive after the system has been running. 'DISPLAY' shows some specific system information. 'ENABLE' and 'DISABLE' control access to certain system information, more powerful (and risky) commands, and the FRONT-PANEL (CTRL-Z). <CTRL-Z> brings up the machine language FRONT-PANEL and allows direct access to the low-level monitor (not for the beginners). The 'SQUEAL' and 'DONT' commands, which are in early Exec's and in Exec/{A|S} instruct Exec to report (or not to) single density disk errors as they occur.

Those are the things the Poly waits for you to tell it to do. How does it figure out what to do? Well, it looks at what you type in. It first checks to see if you have typed in a command. If so, it goes ahead and does the command. If what you typed isn't a command, the Poly assumes you want to run a program, so it looks for a file with the name as you have typed it. If it is a runnable file, it goes ahead and

runs it. If the file is not runnable, it assumes that it is a file of commands and enters command file mode to execute the commands in the file. If it didn't find the file, then it tells you "I can't find that file."

Whenever it can, the Poly translates all known error messages into human readable text.

There are two ways for a beginner to start using the Poly. One way is to use the WordMaster system; the other is to give commands to the operating system (from the '\$' prompt). The WordMaster system is limited to word processing operations, but is easy to use because of its menu-driven organization. Using the Poly Operating System requires learning what the possibilities are and which ones you will want to try. You needn't master all the commands to get useful work from the Poly. With just 'EDIT', 'TYPE', 'PACK', 'DELETE', 'COPY', 'RENAME', and 'PRINT' you can get a hell-of-a-lot of mileage out of a Poly as a simple word-processor. A sports car can be driven only to market.

If anyone out there still considers him or herself a "beginner" feel free to drop me a line or call asking any questions at all. Be anonymous if you must, or request anonymity. But, as I say to my classes, "The only 'dumb' question is the un-spoken one."

### **BugNote**

HELP BUG Sio-1

Abstract Systems BugNote 014.0 January 24, 1983

Sio.PS

Sio.PS has a bug in the kill routine. Whenever the printer driver software connects a new printer driver, or when it re-connects the printer driver, the "disconnect" function is first called by Prnt.OV before loading the new driver (or a fresh copy of Sio.PS). With handshaking, or slow printers, the transmit buffer may still have characters in it waiting to be sent by routines which are active at the interrupt level. The kill routine does not check to see if the buffer is empty (that all characters have been sent) before shutting down the USART. This results in the failure to print a few characters when

a command file is used to restore the printer driver. The following is a command file sequence in which the bug shows up.

```
$Pr SET
Lines per page (form size)? 66
Characters per line (page width)? 132
Lines for TOP margin? 3
Lines for BOTTOM margin? 3
Offset for left EDGE? 6
$PRINT a-text-file.TX
$PAGE
$Printer Normal
```

Usually, the PAGE command is lost. Sometimes, part of the last line of a-text-file.TX is also not printed.

To avoid this bug, do not restore the printer driver using the Printer Normal command, or do not restore the Printer driver from the command file. I have adopted the practice of leaving the last RETURN out of the command file. To finish the Printer Normal command, I must hit RETURN from the keyboard. This allows enough time for the remaining characters to be sent before the USART is shut down by the disconnect routine.

---

### **Advertising**

Commercial advertising rates are \$50 for a full page, \$25 for a half page, and \$15 for a quarter page. Anything smaller is \$3.00 per column inch. A column is 3-3/4 inches wide by 10 inches tall. A full page is 7-5/8 inches wide. Noncommercial ads by subscribers are free.

---

Entire PolyMorphic System User Manual, System 88 User's Manual with Exec/96 addendum, & System 88 Operation Essentials On IBM disk. Al Levy, 516-293-8358

---

FOR SALE: Poly 8810 box with power supply and mother board. \$50 plus shipping. Charles A. Thompson, 2909 Rosedale Avenue, Dallas, Texas 75205-1532, (214)-368-8223.

---

### **DISKS - MODEMS - PROMS - SOFTWARE - SPELL**

1. MAXELL diskettes: 5-1/4" 10 hard sector -- \$15 per box.
2. Used diskettes: 5-1/4" 10 hard sector -- \$0.50 each.
3. Hayes Micromodem 100 (300 baud 5-100 internal modem) \$30.
4. HayesSys modem software (for the Micromodem 100) \$30.

5. Abstract Systems Exec (Enhancements & bugs corrected) \$35.
  6. Abstract Systems Proms (Enhancements & bugs corrected) \$35.
  7. PolyGlut Library Volumes: \$6 each; 5 or more - \$5 each.
  8. Hayes Smartmodem 1200B (IBM compatible internal) \$60.  
Abstract Systems, etc., 191 White Oaks Road,  
Williamstown, MA 01267, Phone: (413) 458-3597
- (Send \$1.00 for a complete catalog--[free with any order].)  
(Make check or money order payable to Ralph Kenyon.)

### Bit Bucket

There once was a bit,  
who lived in a byte  
with seven other bits.  
The space was tight,  
the digits all hit,  
but the number was right!

### Hints

According to John R. Neal, standing the MS drive units on the right side makes them work much better. This takes the weight of the heads off the media.

### The Other Guys

Poly's COMP-DISK program compares two disks. On the PC and compatibles, the program DISKCOMP.COM does the same.

### In This Issue

Editorial . . . . .	1
Open Letter to Quaker Oats . . . . .	1
Announcement - Emulator Upgrade . . . . .	1
Letters . . . . .	1
Benchmarks for the Poly Emulator. . . . .	2
Evaluating Benchmarks . . . . .	3
PM BUG -- Emulator bug . . . . .	4
Printer Sharing . . . . .	5
Primitive Displays . . . . .	6
HELP COMMAND TYPE . . . . .	7
HELP COMMAND UNDELETE . . . . .	7
For Beginners . . . . .	7
BugNote 014.0 - Sio-1 . . . . .	9
Advertisements . . . . .	9
The Bit Bucket . . . . .	10
The Other Guys . . . . .	10
Hints . . . . .	10

**PolyLetter**  
191 White Oaks Road  
Williamstown, MA 01267  
(413) 458-3597

Address Correction Requested

FIRST CLASS MAIL



Ralph E. Kenyon, Jr.                      EXP:99#9  
Abstract Systems, etc.                      184  
191 White Oaks Road  
Williamstown, MA                              01267-2256

© Copyright 1989 by Ralph E. Kenyon, Jr.

PolyLetter Editor and Publisher: Ralph Kenyon. Subscriptions: US \$18.00 yr., Canada \$20.00 yr., Overseas \$25.00 yr., payable in US dollars to Ralph Kenyon. Editorial Contributions: Your contributions to this newsletter are always welcome. Articles, suggestions, for articles, or questions you'd like answered are readily accepted. This is your newsletter; please help support it. Non-commercial subscriber ads are free of charge. PolyLetter is not affiliated with PolyMorphic Systems.

Back volumes of *PolyLetter* (1980 thru 1988) are available at reduced prices payable in US dollars to Ralph Kenyon. 1 - \$15, 2 - \$28, 3 - \$40, 4 - \$50, 5 - \$59, 6 - \$67, 7 - \$75; Canada add \$3 shipping, Overseas add \$10. Individual back issues are also available (\$3.50, \$4.00, \$5.00).



# PolyLetter



PolyLetter 90/2

Page 1

MAR/APR 1990

## *Editorial*

I received a very long letter with many questions in it. Since most of the questions are of interest to our readers, I am devoting a large portion of this issue to the letter and my reply. If any of you have things to add to the discussion, please write in.

Also new in this issue is a printed parrot. Al Levy was kind enough to scan in the PolyLetter delivery bird. (The ferocious one in whose talons your Label is held each issue.) Mr. Squock must still be cut out and pasted on. I was able to print the bird using WordPerfect. Unfortunately, WordPerfect still doesn't know my printer well enough to give the nice justified proportional typeset appearance you have been used to. WordPerfect wants to insert literal spaces like FORMAT does and make irregular spaces between words. I'll work on it in the future.

Speaking of Mr. Squock and the big bad bird, we need a name for the other bird. Send in your suggestions.

## *Letters*

Dear Ralph,

Mar 28, 1990

First of all, our letter to you of earlier this year has not apparently reached you. This is probably due to confusion at our end. We are a designated training route for the U.S. post office, where they train people who are less than adequate, i.e. already trained but needing more training. I got your note two days ago and will send it and check today. I look forward to hearing from you soon.

Enjoyed your letter and listings of what is available. I feel glad to have such a resource to aid me in having Poly up and running. I apologize for any questions that may seem primitive. My understanding

is indeed primitive.

My check of 18.00 for a subscription renewal is enclosed. I will order more when my needs are clearer. My first need is also a new 89/4, PolyLetter, while mostly readable, came apart or was abused by the U.S. Pretzel Service. I would like a more complete copy for my records.

I have been using a "difficult" copy of MAILIST, which is tricky to say the least. It does not pass Sniff or Checksum and we still use it anyway, have for years as a matter of fact. I just hope we never need to use those sectors that are not functional before we get a new copy. Is another copy or new Maillist program available or if there is a version of it which you have upgraded? Or could we just get a copy of it from someone? Ours is the Exec 95 which came with the Poly when we bought our last upgrade.

I am sending this letter on disc along with a print out because I thought that might simplify your response. You may just edit my files with your answers or write a note.

Just finished reviewing PolyLetter from my first issue, in 82 till now. It was joyful, exhausting and amazing. We Poly owners have really been involved in history. We own the first and best of the first generation of home computers. While not a programmer from the beginning, today I feel really privileged to be involved and done so much to keep things running through all the difficulties of low support, finding software and generally being capable to solve simple problems.

I find I do not respect those who have a down computer and no idea what to do but take it in only to find out that the power cord was loose or some such silliness. We know the systematic approach to get our Polys up and running from a very elementary

point of view. Some of the PolyLetter stories of fixing a downed Poly leave me feeling awed with their levels of expertise.

As someone who trained first on the IBM 1401 series, a 16 K monster which with peripherals: tape drive, card reader punch and printer (itself the size of a large freezer chest) which would fill most double garages, I appreciate the progress. That monster was incredibly primitive, but in 1965 we thought the IBM was wonderful, especially those who had to work on the previous generation machines.

I have just finished a review of my Poly letter files, all but the first few issues, just for memories sake and to prepare for this letter intelligently. I have been a resident of PolyWorld for some 8 years now. Due to a lack of technical support, it has been rough at times. I have gotten a portable for my business to carry extensive files. I prefer the Poly and plan to keep it running.

I just read Frank Stearns article from POLYLETTER # 82/5 on the Poly being like a the 57 chevy or the DC-3 in ten or fifteen years, an absolute classic. Today that is only a few years from now. I plan to have my Polys running.

My suggestions for improving PolyLetter are to put out the basics from time to time, such as either an index to how-to articles or a regular series such as has been started several times.

It might be nice to have handouts, pamphlets to aid people in doing upgrades, like a disc upgrade handout, similar to the excellent disc drive care and maintenance series in past PolyLetters.

I am a Poly owner using four working computers. We have original style drives (90k), 8813/8810 (upgraded) with several 75s, a 76 and an 81 Proms, using a Exec version 95 WordMaster or an old MailList (73 or 76?) for all applications. On CPU's we have mostly F's with two 0.3's and one E. For Disc Drives we have B's, one D and three E's.

The following listing represents some of my main questions, starting with a need for several new Proms:

For the Proms, I want whatever will support an 8 inch upgrade. I am not particularly interested in fast ones nor in highly sophisticated ones that require many modifications.

Which is the simple way to increase my disc storage the most effectively in the 8 inch format?

I want to expand to a 7 drives for one machine to sort through all of my discs. I am very unsure how to proceed. I want to know the disc expansion capabilities and requirements for changing SSSD TO DSSD OR DDDD and which of these conversions is needed to use the 8 inch floppies or MS.

As I remember it, one conversion, most likely to DSSD, took only different drives and doubled the storage space. Then there was the change to SSDD which took ROM changes, board changes and new discs which I thought was also needed for the 8 inch drives. Are these correct? To get both options, i.e. DSDD, which I think is the 96 tpi situation we would need all three modifications, to the board, to the drives and to the ROMs. This does not seem necessary to me. If possible, I would just like to get the larger format 8" drives and add them to an existing 8813 system. What does this entail?

On the drive upgrades, it might be fine to have them on single sided discs as the need is not for cramming as much in as possible onto a disc but to simplify sorting of a number of alphabetical files of some length that are presently out of sequence.

Here are a number of questions regarding the programs you offer. DX questions: How do we copy DX files other than manually file by file? I will have to do this soon on many, many discs. The main project involves copying 130 separate files of ten items each from single density floppies to a larger mass storage. Also I will want to copy whole discs as subdirectories onto the larger disc drives when they are installed. I want to make this as simple as practical. Do you have such a program or is this already built up into Abstract Systems Exec?

In PolyLetter Nov/Dec 88, p 2, you talk of DIRCOPY.GO. in an article on moving DX files, seemingly within a disc for across



disc movements at a later date. On which type of disc is this file located? I assume this will move a DX file along with whatever else is on that disc, is that correct? Can it be used selectively, which I doubt from the title itself.

#### Abstract Systems Exec questions:

First of all, is this a Basic disc or is this a WordMaster Disc? We use Wordmaster almost exclusively and so that is needed for our usages. I will probably be ordering AS Exec when we get the 8 inch drive upgrade handled or dismissed, probably mostly to copy DX files if this is needed.

I hope to upgrade two of my systems to 8 drives (I want a back up) and I understand that your Exec would be compatible with this change.

Abstract Systems Exec, seems to need no manuals. You also offer a used manual, at what cost? I would like the manual if it would answer some of my questions.

Double column: I have a large inventory listing of library books that I want to print in a double column mode. Is there a simple poly program to print Exec files in two columns from a non-Wordmaster file (we used tabs for spacing)?

Italics, do you have a program that enables us to print in italics on any printer or just on the IDS Prism?

Modem program: My purpose would be mostly to communicate to my office when I am away as my hearing impairment does not allow me to use the phone. Which is my best FTP program replacement, if any for basic modem communications? I have various programs with the clones but want one with the Poly. Ideally, I would like one that is typical of the P.C and so reduce the learning needed for myself and others. I do not like the idea of having to put in a jumper, so I assume it would be FTP-Hayes? I seem to remember more modem programs for the Poly in the index of PolyGlott Library but not sure of what that all means. Not knowing which is best I am assuming the Procomm from PolyLetter 89/2 is probably best. Do you feel this is correct?

Am I correct that your MICROMODEM-100 is an internal modem or is it just a board to

aid in the communications rather than to use the serial port? Because of the auto-dial function, I assume it is a full modem. FTP Hayes or Procomm would work best with this?

Inventory control, I am assuming that the Inventory listings in Basic would have totals per room, totals per category and/or a grand total. Is this correct? Which would be the easiest to use? I am assuming this is something I would like to do on Poly rather than the Clone. PolyGlott V-17 seems to have the latest version, is this "better" than for instance the programs on V-02 and V-06 PolyGlott library?

On to PC type questions: I now have an use my PC using Microsoft Works for a minor Database with client notes. I dislike the program. What I want is a Poly type word processor that gives me a Wordmaster like screen or a simple \$ type prompt which I can just type into like Poly. What I find is a bunch of word processing bells and whistles that make something so simple on my Poly into a nightmare of different keystrokes. I wonder what the best emulation of the Poly WordMaster is for those Poly owners who have had to switch.

Frank Stearns had an article about MicroSoft Word, which seems to answer my question but I want further responses if appropriate.

If there is any problem in this matter, please call my wife, Carol, collect, preferably in the early afternoon our time.

Thank you for your consideration and time in these many matters.

Here is hoping for some wonderfulnesses in your life, -- Richard Wagner, Dallas, TX

Dear Richard,

Thanks for the long and wonderful letter. I hope I can answer all your questions. But first, here's one new "pretzel" on the way. Thanks for renewing.

MAILIST - I don't think it was changed by Poly in years. It was just re-saved in the new BASIC format each time. Let me know the Version number it signs up with and I'll see if I have a copy of that version; I'll put it in BASIC C04 or C03 if you wish.



I've actually seen an IBM 1401 back in 1966 at Miami University when I was a student. It was in the back room even then because the School had begun to use the (then) new IBM 360.

I've occasionally put some time into working on a comprehensive index for PolyLetter. Special indexes to articles on hardware and software sound like good ideas. I'll work on it.

Most of Poly's upgrade documentation was in the form of ECN's (Engineering Change Notices) and often included step-by-step modifications to the system. However, one needs to be fairly savvy with hardware in order to figure out which ECN's have been installed in any particular machine or on any particular board. There were also some ECN's which are not "serial". For example, there were some early ECN's which were incompatible with each other. When it comes to upgrading a system with one of these early ECN's installed, a technician needed to figure out how to install a later ECN by using steps which do not exactly match those specified in the documentation. If this sounds difficult, that's because it is. Some of them I haven't got figured out.

Adding to the problem is the fact that I don't have all the ECN's issued by Poly. Many were issued with no explanation at all of the purpose. In most cases the boards worked better with the ECN installed, but the good news is that most work without the ECN installed. The Field Service Manual had many of the ECN's in it.

In regard to system proms, here's the scoop. Poly's first system proms were version 75, which basically just added Disk I/O routines to the 4.0 Monitor used by the original Poly-88. These proms were in the first 8813, which had the capability of running up to 3 5" SSSD drives.

Shortly after introducing the 8813, Poly developed their 88-MS 8" drive units for Mass Storage (MS). The MS used the larger 8" drives and the new double density MFM data encoding method. The drive also had a higher data transfer rate, and required more precise timing than the 8080 was capable of. Poly's solution was to give the 88-MS its own intelligent controller with a Z-80 on board. While the Poly Does NOT support DMA, the new double density

controller achieved the same effect by taking up part of the Poly's address space for its Disk I/O processing. The 8080 would copy data onto the Double Density controller card and then wait for a signal from the controller that data transfer was done. To accomplish this, the Dio code in the ROMS had to be re-written. Version 76 is that first version. While they were at it, Poly also changed the SD code to allow for two sided single density 5" drives. But, the disk controller must be modified to take advantage of this. All three single density drives in the system must be either single sided or double sided.

To add an MS unit to a Poly 8813 with PROMS version 76, all one must do is to plug the new controller card into the Poly, and plug in the drive unit. It had its own power supply, so must also be turned on. Two such units can be daisy chained together making drives 4 thru 7.

Proms version 76 is the first version for SSSD plus 8" MS, or DSSD plus 8" MS. The new version 76 roms look to see if the controller card is present; if so it tries to boot of drive 4. If that fails it goes back to drive 1.

Later, Poly realized that the new double density controller could be made to operate the newer 5" drives (which could support double density) with a few modifications and with a new controller PROM for the Z-80. (The double density controller has its own PROM for the Z-80 -- a 2716 EPROM.)

Poly decide that 8813's with these type of drives could also have 88 MS units on them, so the Double Density 5" drives would replace the SSSD drives and controller. A few minor enhancements and bug corrections were made to the system Proms and the new PROMS were version 81. Of course, single density disks can't be read by the double density controller, and vice versa. Version 81 is Poly's final "One size fits all" rom which comes in 2 flavors. One is for SD drives with 8" MS drives. The other is for DD drives with 8" MS drives. Both work without MS units attached.

***All of Poly's PROMS have bugs.***

The SSSD versions all have a "multi sector bug" in the Disk read/write logic so that a bad sector in the middle of a multi sector read or write will not be detected.

That's why your MAILIST seems to run. The bad data is read incorrectly, but because of the multi-sector bug, the roms fail to report the error and, fortunately, the bad BASIC program code is never used. The Sniff command correctly reports the bad sector because Sniff only reads one sector at a time.

The 8" disk read/write logic has a bug in the data transfer logic which opens an interrupt window during critical data transfer processes. A CTRL-Y during this open window can result a locked up system which does nothing.

**ASROM A2 corrects both these bugs,  
as well as adds some enhancements.**

You could add System-88 MS units to any 8813 with proms version 76 or 81, or you could get ASROM A2 proms. The single density controller cards all work with all versions of the roms (except the DD version). Later versions work better because modifications to the disk controller enhanced the operation of the controller. The storage data format was never changed. Poly's MS units can be fitted with single sided or double sided drives. The controller recognizes a signal from the drive to determine which size it is.

When Poly developed the hard-disk system, they created the Volume Manager software, which uses the Vmgr.OV overlay and Driver.DD programs. These software enhancements allowed connecting the hard disk to drives 1 thru 7. The way this worked is that Poly's seven drives, 1-7, became "logical drives". Each drive also had a "physical device" number as well. For example, the first three 5" drives were physical devices 1-3. MS drives 4-7 were physical device 4-7. One hard disk device number was 15. The volume manager assigned a physical device to a logical drive number. For example, my hard disk is divided into several volumes, and one of them is assigned to logical drive 1 (the boot drive).

A side benefit of the volume manager is that additional drivers could be added to allow connecting various types of drives. Poly had one additional Driver.DD which included the code for the SSSD 5" drives. The arrangements provided for by Poly allowed single density 5" drives with an

MS, or double density 5" drives with an MS, but with the ability to add a single density controller and drive.

I enhanced the SD flavor of the 81 version proms and corrected both bugs mentioned. My enhancements were made available as ASROM version A2 (A1 is superseded). I also created versions of ASROM which would take advantage of the fact that later drives could handle 40 tracks and a version which would handle the faster step times of modern drives. Finally, I wrote a version which would handle the 96 tpi drives. Mind you, these are all single density mode. But the drives store more and are faster. I'm told by users that the basic read/write algorithm for the asrom proms is faster than Poly's. The loop for multi-sector reads and writes is more efficient. ASROM's are fully compatible with Poly's SD flavor of version 81 proms, but are free of the Dio bugs in the Poly proms.

I have also developed versions of Driver.DD which will allow booting on SD 5" drives and have the driver code for the double density 5" drives. (Poly only offered a driver which booted on double density proms.)

There are different versions of the single density controller around. Revision E is the latest. The improvements which upgraded from Revision C included a better data separator circuit. The revision C controller will not work on the 96 tpi drives. The higher track density causes too much noise for the old circuit. As a result, the 96 tpi ROMS requires a single density controller revision E. I'm not sure about Revision D. Unfortunately, I don't have an ECN which describes the change.

There are no changes required to the controller cards to switch between roms version 75, 76, 81, and ASROM A2-A5. The only change to the controller is when it is modified from single sided to double sided. And, this only requires cutting two traces and installing two jumpers; it's a very simple modification. By the way, IBM compatible drives can be used in the Poly as is. If the double sided modification to the controller is not installed, these drives would not be fully utilized, but they would do the job. 40-Track ASROMS, together with the double sided controller modification would give 800 sectors per



drive, and allow complete compatibility with Poly's SSSD format.

The 96 tpi ASROM's disk format is incompatible with Poly's standard 48 tpi format. ASROM-96 systems can read the first track (which includes the root directory) of standard Poly disks. To read and write the 48 tpi disks in 96 tpi drives requires skipping every other track, and I have software to do this. The software, called Dio-35, allows reading and writing standard Poly format disks in 96 tpi drives. With the 96 tpi format, one gets 1600 sectors per drive. With my Vmgr.OV, which allows reading double density drives, I get compatibility with Poly's 5" formats as well as fast, large storage on 5" drives.

In regard to sorting... One can sort files spread out on multiple disks, but it might require a lot of swapping. One can use a sort and merge technique. Sort each file then merge two sorted files to a larger output file. I'd need to know more about your operation to recommend more.

In answer to your .DX questions. Poly's program DIRCOPY.GO will copy all files from a drive or from a subdirectory to any other drive or subdirectory. BACKUP.GO does the same thing, but only copies files with the new bit set. (SETNEW.GO and CLEARNEW.GO allow setting or resetting the new bit.) Both DIRCOPY.GO and BACKUP.GO can be told to replace existing output files with the same name. My HELP section covered DIRCOPY.GO in PL 89/6. DIRCOPY.GO and BACKUP.GO both came with Poly's system. Since you have Exec/95, you should already have them. If you don't, let me know and I can get you copies. DIRCOPY is basically for copying all files from a sub-directory or from a drive to another. Suppose you wanted to transfer or copy all files on a 5" disk into a subdirectory named "TEXT" on your MS drive # 5. You could put the 5" disk in drive 3 and then you would use the syntax:

```
DIRCOPY <3 <5<TEXT
```

Suppose you wanted to do another disk into the same subdirectory. When DIRCOPY was done, you could put the new disk in drive 3 and repeat the process. If DIRCOPY finds the same file name in the output directory, it complains and asks if you want to delete it. If you say no, DIRCOPY asks for a new

output file name. You can tell DIRCOPY to automatically delete files with the same name by putting an asterisk on the command line after the output file path thus:

```
DIRCOPY <3 <5<TEXT *
```

When you do this, DIRCOPY automatically replaces any files with the same name. If the files are the same size, DIRCOPY will copy the new file on top of the old one. BACKUP works exactly like DIRCOPY, except it only copies files with the new bit set. I use DIRCOPY to make my first backup of a directory on the hard disk. After than, when I make any changes to any files in the directory, I use BACKUP. Edit, COPY, Scopy, and BASIC will all set the new bit on new or changed files. When BACKUP runs, and it copies (backs up) a file, it clears the new bit.

DIRCOPY.GO makes copying whole disks into subdirectories a piece of cake. Both DIRCOPY and BACKUP create subdirectories as needed on the output drive. But, they both wait until they have copied all files first, which leads me to the article you mentioned.

The article is in PL 88/6, Moving Subdirectories, is "not for the weak hearted". There is substantial risk of destroying a disk when attempting that procedure. The first three paragraphs of that article are all that I recommend to anyone who does not consider him or herself an expert. Here are those three paragraphs for your immediate pleasure.

One annoying property of the program DIRCOPY.GO is that it copies all the files first and then copies subdirectories. The problem with this is that Gfid has to go way out on the disk to get the subdirectory. This adds to the lookup time.

One way to avoid this is to create all subdirectories first and then run DIRCOPY. I have done this by creating a dummy file, X.TX, with only one space in it and then COPYING it from one subdirectory to the next until all the desired subdirectories are created. Then I delete all copies of the file and pack the disk. (I usually create a command file to do the job.)

Now when I use DIRCOPY, all the subdirectories are already created and at



the beginning of the disk, so lookup time is saved. This is easy to do on 5-1/4 inch drives as there cannot be too much data on the disk to copy. On larger disks, however the process may take quite a bit of time. DIRCOPY is much slower than IMAGE. [from "Moving Subdirectories" -- PL 88/6]

Ever see the little sign "Plan ahead" with the last three letters trailing off? When you know you are going to need subdirectories on a drive, whether it be a backup drive or a working drive, it's a good idea to create the directories first. This will insure that they are near the beginning of the disk and will save look-up time later. I recommend creating a couple of spare directories also. Then when you need one, you can just rename one of the spares to the desired name.

Regarding Exec/[A|S]... My system is the complete set of modifications to System-88, the operating system, and includes modified commands. Included are Dfn1.OV, Dfn2.OV, Dfn3.OV, Exec.OV, Gfid.OV, Prnt.OV, and Setup.GO. All errors reported in BugNotes have been corrected in this version of Exec, and new commands (HELP, UnSys, WRITE, and ReStart) have been added. Also, DONT & SQUEAL have been restored and DLIST is recursive for sub-directories.

The System-88 LIST command has been modified to prevent scrolling past protected portions of screen. It knows how many lines are available and only uses that many lines when parts of the screen is protected.

Abstract Systems version of LIST also shows the number of free directory entries, so the program Space.GO is no longer needed. Also, in ENABLEd mode the flags for each file directory entry are shown (D for deleted, S for system, and N for new). With the flags, you can see just which files to use CLEARNEW or SETNEW on before using BACKUP. A ':' is used instead of a '.', and typing ESC will exit (no need for CTRL-Y).

Like the LIST command, the System-88 TYPE command has also been modified to prevent scrolling past protected portions of screen. A ':' is used instead of a '.', and typing ESC will exit (no need for CTRL-Y).

Abstract Systems version of Sniff shows the current sector being tested. It erases the sector number if the sector was ok. It also leaves the last sector number tested on the screen if you exit with a CTRL-Y. Also, you can tell Sniff what sector to start with. If the starting sector is larger than the size of the disk, Sniff checks the entire disk. It defaults to the used sectors.

Sniff d ssss

will start checking with sector ssss on drive d. With this enhancement, the program SNIFFALL.GO is no longer needed.

The "UnSys" command allows one to designate all files in a directory as 'non-system' files. Like "SetSys", it prompts for the directory name, which may be a drive number. "UnSys" then clears the system bit on all files in the specified directory. The system must be in the ENABLEd mode.

The ReStart command re-starts a command file which has been interrupted or aborted. It restores the command file mode flag and allows continuing the command file with the next command. (ReStart cannot restore programs which take control of the CTRL-Y interrupt.)

The "WRITE" command sends text direct to the printer. This command affects only the printer which is selected. "WRITE [text]" prints [text] and a carriage return on the selected printer.

The "SQUEAL" command sets a flag which tells Exec to report single density disk errors. Whenever Exec is invoked, any previously unreported disk errors are displayed:

Error 102: 0002

Error 103: 0004

Error 104: 0001

The "DONT" command resets a flag which tell Exec not to display disk errors. "DONT" cancels the action of "SQUEAL".

The HELP command searches for a text file of help information and displays it on the screen. The help files may be on any drive. "HELP [name]" displays the help file information in file "<?<HF<name.HF". - Example: HELP COMMANDS. - (DEFPATH is not changed by HELP.) "HELP [dir] [name]"

displays the help file information in the file "<?<HF<dir<name.HF" - Example: HELP COMMAND LIST - Custom help files can be added by the user.

In answer to your explicit question, my system can have BASIC, or WordMaster added to it. You will probably want to install one of each for different purposes. What works best for you will depend upon how much of the operating system you need to have "on-line" and whether you will be using the MS drives primarily for data or as the system drive. If you want to boot on the MS drive and have "the entire system" on it, it's best to create custom directories and install the operating system. My suggestion is to create half a dozen sub-directories by the procedure mentioned above and to copy operating system file onto the drive using Futil (which uses SCOPY). My subdirectories are "u" for utilities, "c" for command files, "BS" for BASIC programs, "DT" for data files, "WP" for WordMaster, and "SD" a spare directory [which is now half full of miscellaneous stuff].

Exec uses the following overlays: Gfid, Exec, Emsg, Dfn1, Dfn2, Dfn3, Prnt, & Pack. The BASIC overlays are: Berr, Bslv, Bfun, Bdir, Xref Edit uses Efun. Asmb uses Amsg. The printer driver uses Sio.PS System programs include Setup.GO, Edit.GO, BASIC.GO, Asmb.GO, and WPS.GO

To optimize the speed and storage of your system you should decide how it's going to be used. For example, if you never do any assembly language programming you could omit Asmb and Amsg. If you always use BASIC programs, you should put the BASIC overlays on ahead of Exec.

In any case, Gfid should be the first overlay copied onto the disk. "Gfid" stands for "Get File Identifier" and is used by almost everyone else to lookup files. It saves disk steps if Gfid is almost the first thing on the disk. What gets used most next depends upon you. If you spend lots of time in Exec, copying, listing, deleting, etc, then Exec's files should be next. But, if you go right into a BASIC program and hardly ever use Exec, then the BASIC overlays should be next. If you are a dedicated WordMaster user, then Exec's files should be next, followed by Efun and the WP sub-directory. Once the overlays are installed, then you should put

on Sio.PS, and any other printer driver modules. Finally, install Edit, BASIC and Setup. [Take heart, you only need to do this once.] Make a backup of your installed system and put it away for safe keeping.

The rest of the programs you may wish to install would depend upon your needs. If you're like me, you'll want practically every program ever written on the system drive! I can do it on my Hard disk, but it would fill up a double sided 8" disk and then some. Well, you choose what you want, and how often you will be using it. Put the ones used most often on first to save lookup time. But, it hardly matters on 8" drives; they go pretty fast. The savings is in wear on the stepping system.

In regard to manuals... Everything in Exec/{A;S} works as described in Poly's Manuals, but better. I have created a help file for every command and program in Exec, and have included help files for all enhancements. The information is available on-line just by typing HELP from the Exec "\$" prompt (provided you have the help disks in some drive on the system). The used manuals I have are Poly manuals, which you most likely already have. The manuals listed in PL 88/3 are still available at the prices listed -- note shipping charges to cover mailing costs. Chances are you already have most of them.

#### **Double Column Printing**

It would be a simple matter to write a BASIC program to read the file and put two lines together. It would depend upon whether you wanted to alternate left to right or to split the file half and half. Your printer driver would need to be set up to handle the long lines, or they would be truncated. On my printer I can select 17.5 characters per inch. Then 8 1/2 inches will fit 148 characters per line. First set up the parameters:

```
$Pr SET
Lines per page (form size): 66
Characters per line (page width): 140
Lines for TOP margin: 3
Lines for BOTTOM margin: 3
Offset for left EDGE: 8
```

If you don't do this, Sio will truncate long lines. Here are two programs to print in double columns.



Here's a left and right program.

```
10 DIM L$(1:64),R$(1:64),F$(1:31) \FILE:2,LIST
20 INPUT "File to print?",F$ \FILE:4,OPEN,F$,INPUT
40 READ:4,L$ \IF L$="" THEN PRINT "Back to ",CALL(0)
60 READ:4,R$ \PRINT:2,L$,CHR$(9),R$ \GOTO 40
```

Here's a half and half program

```
10 DIM L$(1:64),R$(1:64),F$(1:31) \FILE:2,LIST
20 INPUT "File to print?",F$ \FILE:4,OPEN,F$,INPUT
30 FILE:5,OPEN,F$ \INPUT "How many items in the file?",N
40 FILE:5,POS INT((N+1)/2)+1
50 READ:4,L$ \IF L$="" THEN PRINT "Back to ",CALL(0)
60 READ:5,R$ \PRINT:2,L$,CHR$(9),R$ \GOTO 50
```

### *Italics*

In regard to ... First, the printer must be capable of printing italics. I only have italics on my Data Products 8050 printer. The IDS prism is not capable of italics. I have written code into format.GO which recognizes the {ital} and {ital} commands and which outputs the correct control codes for the DataProducts 8050. On other printers, {ital} defaults to {wul}. I could modify this to include the italics codes for other printers, but would need to know what those codes are.

Modem programs... Procomm is a PC program; it does not run on the Poly. We have used Procomm on PC's while talking to a Poly. Programs which run on the Poly depend upon what kind of modem hardware you have. FTP.GO will run an external modem, but is a plain vanilla program designed to talk to FTP.GO. It doesn't dial and can't save ascii files. It can only send and receive to another Poly running FTP. FTP-Hayes is a modification which uses the Hayes MICROMODEM-100. Both programs can talk to other systems in a straight terminal mode. The program is designed for half duplex mode, so echos its own outgoing characters and must be modified for use with systems that provide echo characters.

The MICROMODEM-100 is an internal modem which plugs into the Poly's backplane. The advantage of any internal modem is that it wouldn't use the serial card interface. External modems use a serial port so must take the place of the printer. If you are lucky enough to have two serial cards, one can be dedicated to the modem, but there is only one serial port in the Poly, and if the modem were connected to it, the printer would be disconnected. The advantage of the MICROMODEM-100 is that it doesn't take away the ability to use the printer while

using the modem.

HayesSys allows using the MICROMODEM-100 to send and receive ascii files, to log the incoming data to the printer, and to place the call by dialing for you. With a MICROMODEM-100 and HayesSys, you could sit down to your Poly and type in CALL <number> and it would dial the number and wait for the modem link to be established. Then you would on-line with the receiving system and could operate it just as if you were in the office. If you wanted to receive some stuff from the office you could set the Poly to log the incoming to either the printer or to a disk-file. You could also send stuff to the office computer (in ascii mode).

If you had an external, Hayes compatible modem, then Bob Bybee's SM.GO program could do the job, but without the ability to use the printer. Bob's program also supports XMODEM file transfer protocol. The MICROMODEM-100 is also limited to 300 baud. I do have one U.S. Robotics 1200 baud internal modem which I have used and wrote some software for. The tradeoff seems to be one of speed vs price. If you buy a MICROMODEM-100 and use HayesSys you will get off cheap, but the transfer speed will be slow.

Inventory... PGL-V-17 is an upgrade of PGL-V-02. PGL-V-06 is something written by someone other than Poly. I do not know much about any of them.

PC questions... There are many public domain or shareware word processing systems for the PC which don't have a lot of bells and whistles. One which has the edit program and the print program separate is PC Write. It has several bells and whistles. A very simple editor is called FRED (Free Editor). It doesn't even have Poly Edit's features.

Let's hear from our readers on what they use for editing on the PC.

---

### *Advertising*

Commercial advertising rates are \$50 for a full page, \$25 for a half page, and \$15 for a quarter page. Anything smaller is \$3.00 per column inch. A column is 3-3/4 inches wide by 10 inches tall. A full page is 7-5/8 inches wide. Noncommercial ads by

subscribers are free.

PolyMorphic 8813 needs home. Make offer. Conway Spitler, P. O. Box 385, Fillmore, CA 93016-0385

Entire PolyMorphic System User Manual, System 88 User's Manual with Exec/96 addendum, & System 88 Operation Essentials On IBM disk. Al Levy, 516-293-8358

FOR SALE: Poly 8810 box with power supply and mother board. \$50 plus shipping. Charles A. Thompson, 2909 Rosedale Avenue, Dallas, Texas 75205-1532, (214)-368-8223.

DISKS - MODEMS - PROMS - SOFTWARE - SPELL

- 1. MAXELL diskettes: 5-1/4" 10 hard sector -- \$15 per box.
- 2. Used diskettes: 5-1/4" 10 hard sector -- \$0.50 each.
- 3. Hayes Micromodem 100 (300 baud S-100 internal modem) \$30.
- 4. HayesSys modem software (for the Micromodem 100) \$30.
- 5. Abstract Systems Exec (Enhancements & bugs corrected) \$35.
- 6. Abstract Systems Proms (Enhancements & bugs corrected) \$35.

- 7. PolyGlott Library Volumes: \$6 each; 5 or more - \$5 each.
- 8. Hayes Smartmodem 1200B (IBM compatible internal) \$60. Abstract Systems, etc., 191 White Oaks Road, Williamstown, MA 01267, Phone: (413) 458-3597 (Send \$1.00 for a complete catalog--[free with any order].) (Make check or money order payable to Ralph Kenyon.)

**Bit Bucket**

Some claim that "bit" stands for binary digit.

**In This Issue**

Editorial . . . . .	1
Letters . . . . .	1
Reply . . . . .	3
All of Poly's PROMS have bugs . . . . .	4
DIRCOPY.GO . . . . .	6
Double Column Printing . . . . .	8
Advertisements . . . . .	9
The Bit Bucket . . . . .	10

**Coming Soon**

Managing TSR programs, Poly Meta, More PC stuff, A review of PC Outline.

**PolyLetter**  
191 White Oaks Road  
Williamstown, MA 01267  
(413) 458-3597

Address Correction Requested

**FIRST CLASS MAIL**



Ralph E. Kenyon, Jr.	EXP:99#9
Abstract Systems, etc.	184
191 White Oaks Road	
Williamstown, MA	01267-2256

© Copyright 1989 by Ralph E. Kenyon, Jr.

PolyLetter Editor and Publisher: Ralph Kenyon. Subscriptions: US \$18.00 yr., Canada \$20.00 yr., Overseas \$25.00 yr., payable in US dollars to Ralph Kenyon. Editorial Contributions: Your contributions to this newsletter are always welcome. Articles, suggestions, for articles, or questions you'd like answered are readily accepted. This is your newsletter; please help support it. Non-commercial subscriber ads are free of charge. PolyLetter is not affiliated with PolyMorphic Systems.

Back volumes of PolyLetter (1980 thru 1988) are available at reduced prices payable in US dollars to Ralph Kenyon. 1 - \$15, 2 - \$28, 3 - \$40, 4 - \$50, 5 - \$59, 6 - \$67, 7 - \$75; Canada add \$3 shipping, Overseas add \$10. Individual back issues are also available (\$3.50, \$4.00, \$5.00).



# PolyLetter



PolyLetter 90/3

Page 1

MAY/JUN 1990

## *Editorial*

More than once a reader has suggested that the content of PolyLetter was too technical. I would be happy to include articles and discussions of a less technical nature; but, no one has recently sent me any, nor suggested specific areas they would like to hear about.

Topics which were requested and have been published include; assembly language article governing the use of WHO, WH1, CKDR, MSG, etc; what happens when BASIC is saved with SAVEF, or SAVEP; UNSAVEP; how would CP/M be of use; Front Panel; assembly language; drive service; keyboard service; hardware update recommendations; source lists; communication software articles; file transfer to other computers; file interaction with non-Poly machines; more on PC clones; more articles on transitioning to PC software; more articles on hardware (boards, etc.); more articles on languages; relocatable files.

Many of the requested topics were of a more technical nature. Perhaps those of you who would like to see some less technical articles would write in with the topic areas of your interest. I can serve you best if you tell me what you'd like. Write or call.

I've added the masthead to the portions of PolyLetter which were scanned in and are printed graphically. This issue has a review of PC Outline and my revised SLIST.GO program. A couple of issues ago I sounded off on recycling. The response was encouraging. Here I go again.

Recycling is good, but it's not the answer. Our real priority ought to be to stop the production of waste at the source and to only produce reusable products. Toward that end I have begun notifying junk mail sources with the following letter which I urge you to adapt for your own use.

Greetings,

As you are aware, our environment is now seriously threatened by global warming, increasing pollution, and the decrease of ways and places to dispose of waste. Recycling is not the answer, but merely a short-term stop-gap measure that does not address the cause of the problem, but attempts only to treat the symptom. The real cause of the problem is the production and consumption of material which is destined to become waste after only one use, or without being used at all. We need to educate our ways of looking at things and our ways of living and to begin to produce and consume only products which fit in a continuous and renewable use cycle. Toward that end I am examining all products received with the view to terminate, at the source, any which fall into this category. The vast majority of mail-order catalogs fit into this category.

Unfortunately, the catalog you send to me fits into the category of non-use or minimal use. I have not ordered from you recently, and do not anticipate doing so in the foreseeable future. Please code my name on your list as a person or firm which is not to receive any advertising or other promotional mailings. This will save you the expense of sending me the material, me the expense of disposing of it, and the environment the expense of its production and disposal. If and when I need products such as you advertise, I will contact my local reference librarian for a list of producers of the desired product and will solicit a one-time-only catalog for those future needs. In the meantime, it is more important that we begin to reduce the production and distribution of products which have only a limited life before they become waste.

My name may appear on your list in any of the following forms: ...  
The affected address are: ....

Please code any variations on these names and address to be excluded from all mailings.

Sincerely,

By the way, if you print your own labels, you can use the return envelopes sent with some junk mail for your own purposes. (No, it's not a sign of being "cheap"; it's a sign of awareness that every little bit of energy saved contributes to stopping global warming!) But, be careful to cross out or cover up any bar-code. I sent out one of these envelopes and the Post Office sent it to the bar-coded address instead of the printed one!

### Letters

PolyLetter

June 7, 1990

Frankly, most of PolyLetter content is "over my head" or has little relevance. I do not want to convert to a PC, but in that case you would be most helpful. I have 4 8813's and 3 hard disks. Can fix nearly all problems. -- James Salinger, Cincinnati, OH.

Frankly, I'm happy to hear from someone who has no intention of converting to a PC. I expect to use my Poly for a long time to come. Unfortunately, many readers are converting to the PC and want to hear about "painless" ways to get there. The Poly and System-88 will remain PolyLetter's dominant theme for some time to come, but we will provide some PC conversion support too. As far as the Poly goes, it's still my first love. You mentioned fixing things. Please take some time to jot down some of the problems you have fixed and how you fixed them. Share the result with the rest of us (see the next letter). -- Ed.

Dear Ralph,

June 29, 1990

I'd like to see more pictures. Also, what are common causes of breakdowns of hardware, other than poor connections (which can be fixed by cleaning) and failed voltage regulators? -- Karl Thomas, Elk Grove, IL.

Dear Karl, -- Other than drives going out of alignment, media wearing out, and an occasional memory chip getting fried by static electricity, I've had only one failure that could not be traced to either

a failed voltage regulator or poor or dirty connections between either the boards or the chips. That problem was with my Printer mini-card. One of the chips went bad. When the printer card was selected, it would throw the system into the front panel. One such failure was the 8T97 chip, which can be replaced by a 74LS367. Another failure was either the 1588 or the 1589, but I don't remember which. The serial test plug and the SERIAL-TEST.GO program make trouble shooting this board quite easy. But, enough of my ramblings. Let's hear from our readers. What kinds of problems have YOU had? People want to know, and how to fix them. -- Ed.

### slist.GO

Lately, I have begun to have problems finding files on the Poly. I am running a hard disk with many sectors and several volumes. Each one of the volumes has many subdirectories, so finding a particular file becomes a chore now.

I have also been using my XT clone more lately, and I like the convenience of the flexibility of the DOS DIR command. DIR can take as arguments wild cards in both the file names and in the file extension. "DIR \*.TXT", lists files with "TXT" extension. "DIR name" lists all files with any extension but named "name". "DIR ????.TXT" lists all files with three letter names and the extension "TXT".

I have been using my "slist 1.X" program which produces a list of all files in all subdirectories in sorted order, but it did not allow selecting files. Recently I decided to rectify that and have enhanced slist to allow the same kind of flexibility that the DOS DIR command does. My updated program, slist 2.0 now includes all the flexibility of the DOS DIR command and more. I have been trapping slist's output with the Printer LOG command and Printer File. This allowed me to edit the list to remove miscellaneous junk, and then SORT the list in another order. [See the article PACK-CRASH]. To save all that editing, I decided to give the upgraded slist program the capability of sending its output to the printer. I can also tell slist to limit it's work to a particular drive and subdirectory.

One thing I am often interested in is which files have the NEW bit set for



backing up. I gave slist a switch to tell it to list only new files. It can also list only deleted, or system files, or any combination thereof. The real boon came from the ability to select which files to display. If you give slist a file name it will display only files beginning with the same letters as specified. A question mark will match on any character, just like in DOS. The possibilities will be a bit more apparent if I show you the syntax for executing slist and for reentering the program. The program has 3 parameter fields. Each may be omitted and has a default value.

```
slist %1 %2 %3
slist [d<[path]] [Name:Name.EX:*.EX] [/DSNP]
REE [Name:Name.EX:*.EX] [/DSNP]
```

The first parameter field is the drive or drive and path field which tells slist which files to select from. This parameter may be a drive number, a drive and subdirectory path, or may be omitted. If omitted the default is the entire system drive.

The second parameter field is the select file name. If this parameter is omitted, "\*.??" is assumed. All files will be listed in this case. If the first character is anything but "\*", the file name will be checked character for character for a match until the given characters are exhausted. This has the effect of searching for file names which begin with the given letters. A question mark matches on any character. Here are some examples. The first parameter is omitted, so the following examples all search the system drive.

```
slist Name.EX Searches for an exact match
slist Nam Searches for files beginning with "Nam"
slist ?A Searches for files having an "A" in the second
position
slist *.TX Searches for all files with the extension "TX".
slist ??? Search for three character file names.
```

The third parameter gives the switches and select which files to list and whether to send the list to the printer. The slash identifies this parameter field. The actual parameters may be any combination of "D", "N", "S", or "P" in any order. "P" instructs slist to send the output listing to the printer instead of to the system screen. "D", "N", and "S" select files with the Deleted, New, or System bits set.

The system must be enabled to accept these options. The default is all files in enabled mode, non-system and active files in the disabled mode. In the enabled mode, the switches restrict the listing to those files with the appropriate bits set.

```
slist /NS will list new or system files in enabled mode.
slist /N will list all files which need to be backed up.
slist /D will list all deleted files.
```

All in all, slist.GO has been more and more useful to me as time goes on. It makes finding files on large capacity disks much easier. If anyone wants a copy, send me \$5.

## PACK-CRASH

Well, one of my greatest fears happened. The system crashed while it was packing a large disk with many files. Recovery was a real horror story. The moral of the story is "Backup before you pack original disk". However, with a few tools, I recovered everything. "How'd ya do it?", I hear you ask.

We must first try to backup the entire disk, if possible. This can be done by changing the next disk address field to the size of the disk. When you are looking at the root directory with SuperZap, the next disk address (Nda) is the next to the last and the second from the last bytes on the top line. These are in byte reversed order. Pack will have changed this to point at where the next free disk address is supposed to be when the pack process completes successfully. So, if Pack crashed, it will actually point at some spot earlier than the end of your data. We can set this to the highest possible value for each type of disk drive in order to backup the entire contents of the crashed drive (hoping there are no media errors). Here are the value for various disk sizes.

	Drive	type	Size	Hex	Reversed
1.	PM	SSSD	5 350	15E	0E 01
2.	PM	DSSD	5 700	2BC	BC 02
3.	PM	SSDD	5 700	2BC	BC 02
4.	PM	DSDD	5 1400	578	78 05
5.	PM	SSDD	8 2464	A00	00 0A
6.	PM	DSDD	8 4928	1340	40 13
7.	AS	SSSD	5 400	190	90 01
8.	AS	DSSD	5 800	320	20 03
9.	AS	SS96	5 800	320	20 03
10.	AS	DS96	5 1600	640	40 06
11.	IS	SSQD	5 1600	640	40 06

12. IS DSQD 5 3200 C80 80 0C

Drive type number 1, 4, 5, and 6 are the most common. Once you use Szap to change the Nda bytes in the root directory, listing the drive should show the disk as being full. Type 1 drives will show 350 sectors in use; type 4 drives will show 1400 sectors in use; etc. Now, IMAG will copy the entire drive, bad directories and all, unless there was a media error. In that case, you will need to use RECOVER to recover the good sectors. You may be able to fix the bad sector by writing it with Szap, but that won't always work. Let's assume we have our backup (or are risking destroying the data).

To recover from a pack-crash we should understand what Poly does during the pack process. First she reads all directories and builds a list of pointers to contiguous spaces on the disk (or to the holes, depending upon how you look at it). Then she updates the directories by removing delete file entries and updating the disk address locations. Next she writes out the updated directories. Finally, she begins reading and writing the blocks of data which must be moved down to fill in the missing spaces. She is so smart that she can even find out when you have reduced the size of a file -- perhaps with Szap. (This was necessary for the case of a deleted directory.) But, if you lose power before the process is completed, your file directory entries will point at where the files are supposed to have been put in the final process and not where they actually are when the crash happened.

One thing to notice is that the files are on the disk strictly in disk address order. This is true even if the directory entries are scattered throughout different directories. What we need to see where the process went wrong is a complete listing of files, together with their subdirectories. And, this list must be in the order that the files are on the disk. Fortunately, we can get such a listing, but it takes a couple of steps. My slist.GO program reads all sub-directories on a disk and sorts the files by name order. But, in the ENABLED mode, it includes the disk address in the sort. Our first step is to run slist (which I have renamed "sl" since I use it so often) on the offending drive.

Two things can happen. The one we hope

for is that the program will run correctly and give us our sorted list. This will only happen if no directories were moved. The other thing that might happen is that sl reports "Disk directory destroyed!". This is likely to mean that when sl looked up a sub-directory, what it found was the contents of some other file. Of course, the checksum doesn't match because the area pointed at by the disk address isn't where the directory is anymore. Here's where we must do some patching. Record the disk address where the system says the offending directory is located. (Remember, because of the crash, the directory is actually farther out on disk and hadn't been moved down yet.) Now, bring up Szap and set the disk address to the address where the directory is supposed to be. You can expect to see the contents of some other file. Begin scanning ahead with Szap until you find the directory. When you find it, make a note of the disk address where you do find it.

Now we must update the directory entry for the parent directory which pointed to the wrong place. Set Szap to look at that directory and find the directory entry which names the mis-located directory. Move the cursor to the disk address field in this directory entry and change it to the actual address where the directory was found. (Remember that the address bytes are reversed -- lo byte first, hi byte second.) We're not done yet. We need to correct the checksum for the directory we have just changed. Back up with Szap to the first sector of this directory and hit CTRL-C, which tells Szap to compute and replace the checksum. (If you are in the root directory, Szap automatically does this for you.)

After we have corrected directory address, run sl again. If there were more directories moved we may have to repeat the process above. Keep repeating it until all moved directories have been found and corrected and sl runs correctly to completion.

Once this is accomplished we need to save our sorted listing. I did this using Print-to-a-file, which is connected to Printer "File" on my system. Select some other drive than the damaged one, or you are likely to write over the files you want to recover. I suggest using a clean new floppy for the process. Anyway, we must



capture the output of sl into a file. I did it with the following commands.

First I set Print-to-a-file to output to a file called "SL" on drive 2 with the command "Printer File <2<SL.TX". Next, I told Poly to LOG the display with the command "Printer LOG". Then I ran sl on the offending drive. It gave me a listing on the screen in sorted order of the files on the bad disk. (I kept having to press the space bar to keep paging the display). When all files were displayed, I closed the capture file with the command "Printer Null". The next thing to do is clean-up the capture file. When sl stops after a page, it displays a colon on the screen; when you press any key, it deletes the colon and displays the next page. But, Printer LOG captures both the colon and the delete character in the file. So, I Edited SL.TX and removed all the ":■" pairs from beginning of the lines where paging occurred. I also removed any form feeds generated by the printer driver, and any other extraneous lines, such as the "Printer Null" at the end of the file. The result is a nice clean listing of all the files on disk with their subdirectories; unfortunately the list is in alphabetical order by file name -- we need it in disk address order.

The next task is to re-sort the file using SORT.GO. SORT can be told which column to sort on and the disk address starts in column 6. The command

```
"SORT /+6 <2<SL.TX <2<FL.TX"
```

will resort the list into disk address order. Just to make it easier, I printed out a copy of FL so I could refer to it while continuing with the next step.

The next stage in the process is to start looking for misplaced files. We use Szap on the offending drive and scan forward, looking at the disk address and the file on our printed list and comparing it to the address in Szap. Of course, to do this, we must have a pretty good idea of what is in each file on the disk.

Sooner or later we will find a file which isn't located where the listing says it is supposed to be. Make a careful note of the disk address where it is actually found next to the address on the list where it is supposed to be. Cross your fingers

and hope that you are near the end of the list! We must continue this search for all the remaining files on the list; make a note of the actual address where the file was found next to every remaining file. The process can be shortened a bit by using the size of the file to skip ahead to where the next file might be located.

Once you have the complete list of disk addresses which must be changed, we can begin entering the changes. We use Szap to go to the subdirectory where the file is listed; then we change the disk address in the file directory entry to the correct value (byte reversed). Finally we go to the first sector of the directory and compute a new checksum (CTRL-C).

After the file directory entries are updated, we need to TYPE each file which has been changed to make sure that we have actually made the change correctly. There is one special case you may find with the last file which seemed to have been in the correct location. Be sure to check this one carefully. There are only three possibilities. The whole file is moved down and it is intact (not very likely), the beginning of the file has been copied down and the entire file is still intact farther out on disk, or, and this is the harder case, the beginning of the file has been moved down while the tail of the file is still further away. If the first case holds no further correction for this file is required. If the second case holds, we need to change the disk address to point at the later location where the entire file is actually located.

The third case is most likely. In that case you will have to use Szap to find both the beginning and the end of the file. Make a note of the first sector address and the last sector address (as found by Szap). You will need to do some hexadecimal arithmetic with these numbers. Add 1 to the last sector address and subtract the first sector address. This will be the total number of sectors that the split file spans. This time you need to use Szap to update the size of the file so that it includes both the beginning and the end of the file (We'll Edit it later to remove the excess). As an alternative, you could use RECOVER to copy the entire split file to drive 2. If the file was not an editable file, use RECOVER; we'll need to sort out the pieces and reassemble it using another

technique which I'll save for another article.

At this stage you should be able to TYPE any file on the offending disk and see what's supposed to be there, except in one case, you may see some duplication or some extra garbage in the middle of the file.

At this point you should have corrected all the directory entries which were incorrect. You are now ready to cross your fingers and pack the disk again. Providing there are no media errors on the disk, one does not occur during packing, and you don't lose power again, you should get everything back where it belongs. Since I got my hard disk for my Poly, I have faithfully kept up my backup disks. I have a backup disk for nearly every subdirectory on the hard disk. But, as Murphy would have it, I was packing the hard disk before backing everything up, and had several new files I needed to have. I actually caused the problem; I leaned over the computer and bumped the load button while it was packing. But, I recognized with horror what might happen if the power went down during a pack.

I am probably not the first and won't be the last to have this happen. The early Apple computers had a RESET button right on the keyboard and people often hit this, losing their program in the process. But whether you lose power, or push the load button, the recovery process is the same.

### ***The Other Guys***

I've been working on converting a Poly program to GWBASIC. The Poly PLOT command posed one conversion problem. I used the PC LOCATE ROW,COLUMN to replace the Poly PLOT X,Y,F command.

Poly graphics run from 0 to 127 for the X value and 0 to 47 for the Y value with the lower left corner of the screen being the 0,0 point. The third parameter on the Poly PLOT command turns the pixel on or off with a 1 being on. Much of the time the Poly PLOT command is used to locate the cursor before printing a field of text, so the value of F doesn't matter. It is usually written over by the next PRINT command. When PLOT is being used for this purpose, there is a direct conversion to the PC LOCATE command.

The PC BASIC LOCATE command uses the ROW going from 1 to 24 and the column going from 1 to 80 with the top left corner being the 1,1 point. When you are converting Poly PLOT statements to PC LOCATE statements for text positioning you may use the following formulas.  $ROW=(47-X)/3+1$  and  $COLUMN=X/2+1$  where X and Y are the Poly plot arguments. This leaves your plot in the first 64 columns and the first 16 lines of the PC screen. I suggest doing the entire conversion first, and then moving things around later.

### ***PC Outline***

Poly users who are used to using Edit and Format, or WPS, are hard to satisfy when it comes to finding word processors on the PC. I have begun to use PC Outline (PCO), and think it deserves consideration by any Poly user who uses a PC. It is quite different from Edit and Format, but has several things going for it that make it easy to use.

PC Outline is a "what you see is what you get" (WYSIWYG) word processor. It was primarily designed to facilitate writing in the top-down style which starts with an outline. As such it can be useful in writing structured programs, lists with sub-lists, etc. It would be particularly useful in writing complex contracts with all the "but, whereas, and so forth" clauses, etc.

Two big plusses are the ease with which it can be configured for individual printers, and the fact that up to nine (9) windows may be open on up to 9 files at a time, and text can be moved, or copied from window to window. PCO may also be used in a TSR mode, in which case text can be exported and later imported to another file.

All of PCO's facilities are accessible through pull-down menus and sub-menus. Many commonly used facilities are also available directly through the use of control characters or function keys. Here are the control character definitions

^A - find Again	^B - move Block
^C - Copy block	^D - Delete Family
^E - Erase Block	^F - Find string
^G - Insert Toggle	^H - Hide children
^I - Indent	^J - Join elements
^K - Key define	^L - Lose children



^M - Move outline      ^N - New element  
 ^O - Indent and Hang   ^P - Print  
 ^Q - Copy Family      ^R - Replace  
 ^S - Save Outline      ^T - Text Style  
 ^U - Divide            ^V - Center  
 ^W - Move Family      ^X - Unindent  
 ^Y - Delete Line      ^Z - List Windows

Shift Left - Word left  
 Shift Up - Up one outline entry  
 Shift Down - Down one outline entry  
 Shift Home - Top File  
 Shift End - End File  
 Shift PgUp - Paragraph Up  
 Shift PgDn - Paragraph Down  
 Shift DEL - Delete to end of line  
 Shift INS - Insert/Overwrite toggle

As you will note, many of the functions are mnemonic, like Poly's Edit. But, if you forget what any one does, you can always activate the pull down menu and find the appropriate function. Each menu item has the appropriate control character shown so you may refresh your memory and learn quickly the fast access method. Most of the above brief explanations are fairly self explanatory, but some merit more discussion.

Because PCO is outline oriented, it is designed to allow editing whole blocks of outline elements; several commands allow doing this easily. "Hide children" causes the children of an outline element to be temporarily invisible. This allows one to see the outline structure in a more abstract form. PCO also has a "hide-text" feature. Between these two features practically any level of abstraction of a document may be viewed. Let's look also at the function key definitions. The function keys are defined as follows.

#### Unshifted

F1 Help                      F7 Promote Marks  
 F2 Mark current            F8 Indent Marks  
 F5 Promote Current        F9 Copy Marks  
 F6 Indent Current        F10 Move Marks

#### Control + function key

^F1 underline on          ^F6 italics off  
 ^F2 underline off        ^F7 superscript on  
 ^F3 bold on                ^F8 subscript on  
 ^F4 bold off               ^F9 both scripts off  
 ^F5 italics on            ^F10 unexpandable blank

#### Alt + function key

aF2 Clear marks  
 aF9 Copy marks  
 aF10 Move marks

#### Shifted function keys

sF1 static Date            sF2 live Date  
 sF3 static Time            sF4 live Time  
 sF5 doublestrike on      sF6 doublestrike off

Also, the cursor keys operate as follows

Shift Right - Word right

^Home - Top File  
 ^End - End File  
 ^PgUp - Jump to parent  
 ^PgDn - Text hide toggle  
 Gray Plus - Hide Children Toggle  
 Gray Minus - Zoom  
 ^Right - Delete word right  
 ^Left - Delete word left

The Zoom command above changes the size of the current window from full screen to partial screen and back. This is useful when you have more than 1 window open and you want to see the windows in relation to each other. Jumping between windows is accomplished with the Alt key and the window number.

Alt 1-9 - Switch directly to Window #

Some of Poly's ESCape definition features are available in PCO. CTRL-K allows you to define a key. There is no equivalent to Poly's "ESCAPE colon" infinite repeat.

PC Outline is distributed as shareware and can be gotten from the various vendors of public domain software. Mind you, PC Outline is NOT public Domain. You are expected to register your copy if you decide to use it.

As good as PCO is, it is not bug free. I found a couple of bugs that could be gotten around. The serial port support does not work. To print a file to a serial printer, you must select output to a file and print from PCO to a file. PCO gives the file the name of your outline with .PRN as its extension. You may then exit from PCO and use the DOS copy command to print the file.

### **Managing TSR Programs**

TSR programs are programs which Terminate and Stay Resident. The terminology comes from the PC world. We Poly users, who invented Personal Computer

TSR programs, called them memory resident utilities.

The first TSR programs on the Poly were device drivers. Back on the Poly-88 a printer driver utility called "BPRINT" was loaded on top of a portion of BASIC. BPRINT connected routines to allow BASIC to access the serial port. Later, the concept of BPRINT was imported to the earliest Exec as serial device driver capable of driving a couple of different printers. Printer.GO was composed of a resident portion and a transient portion. The resident portion loaded below user memory; the portion extending into USER memory was the transient program. In the latest Exec's, the resident portion is in Sio.PS, and the transient portions has been split up into Prnt.OV and Setup.GO. Prnt.OV also has copies of a part of the resident portion which it installs during boot up. Printer has been made a command in Exec, so accessing the Printer driver "looks the same" to the user in all version of Exec.

But these device drivers all needed to operate in a particular location in memory. Other early device drivers -- Abstract Systems Filer.GO, Poly's Snag.GO, etc. -- also operated in a fixed location in memory. The problem with this is that only one program can live in one location in memory at a time. We would like to be able to connect more than one of this type of program at the same time. The earliest attempts at this took two approaches -- one for tiny programs, and one for larger programs. The tiny program approach took advantage of some free memory on the CPU card. Control-U, which copied the screen to the printer, was one of these. Another is a clock program which keeps the time on the screen. In order to manage these programs, one needed to have the source code and change the load and start address for each program so that each one loaded after the previous one. In some cases one might cut into the system stack and send the Poly into HyperSpace. A variation on this theme called for loading the program into user memory and copying only the small part down into the appropriate location. At one time I had three of these programs all connected at once and all living on the CPU card. Of course, I had to re-assemble all three each time I changed one.

The second approach, for large programs, was to set the load and start addresses to

a high memory area, and then to reset MEMTOP to protect the program. Poly's early debugger program (adb.GO) and Gnomus were examples. But, these programs had to be assembled to load an run at different addresses, depending upon how much memory the user had. They also had to be assembled at different addresses to prevent conflict.

The need to reassemble programs depending upon the user's amount of memory is most cumbersome. Gnomus and adb were not released to the public. Poly kept them for in-house purposes. However, Poly began an ambitious project to update the early menu-driven word processing disk. The first menu-driven WordMaster disk used an overlay to load and execute system overlays, FORMAT, and Edit; but, the more sophisticated Edit and FORMAT required more shared memory to do away with the questions and implement a wholly menu-driven approach. For this project Poly invented the relocatable assembler (See "Relocatable Files" in PL 89/6). The new WordMaster would use several relocatable files which would be loaded into high memory, but which would still allow Edit, FORMAT, or Setup to run in USER meory.

Once the relocatable assembler was available, it became easier to write programs which could be moved up under MEMTOP and protected. The early programs written to take advantage of this simply copied the WordMaster loader and patched it to load .RL files into high memory. No provision was made to disconnect most of the early TSR programs. To disconnect the high memory TSR programs, it was necessary to re-boot the system. FUNCTIONS.RL, and KISS.RL are examples. These TSR utility programs which interface to BASIC cannot be removed except by re-booting.

Abstract Systems did have a ClockOff.GO program to disconnect Clock.GO. But, this program was specifically designed to disconnect one of the tiny programs which lived in the CPU memory area. It could not be used for disconnecting high memory programs. WordMaster II however, was written to disconnect itself and return the freed memory to the user. WordMaster II however, is not strictly a TSR program. It's more of a shell which allows running specific other USER programs. One can't leave WordMaster and have it remain resident. But, WordMaster does load TSR



modules into high memory and unload them as you select menu options.

Poly finally adapted adb.GO to the relocatable philosophy and created RDB.GO, which was distributed with the second release of the System Programmers' Guide. RDB.GO (Relocatable DeBugger) did have a provision for disconnecting itself. Poly's RDB.GO did have a bug in the disconnect routine; it did not reset the single-step interrupt vector upon exit. Aside from this bug, it could be disconnected. RDB returned the freed high memory to the user. I know of no other Poly program which was a relocatable TSR program.

I had rewritten the Clock program to use high memory, but wanted to be able to easily disconnect it. If Clock is running and you try to use Szap.GO, Clock will write on top of Szap's Error data or the top right portion of Szap's display. I wanted to disconnect clock before using Szap. Since Clock was in high memory, I couldn't use ClockOff or even a similar strategy to disconnect it. ClockOff wouldn't know where to find Clock or what MEMTOP was. Also, suppose RDB was loaded after Clock -- we wouldn't want the act of disconnecting Clock to unprotect RDB! I began to think of high memory as an upside-down stack. I would "stack" TSR programs up there one at a time, and remove them in reverse order. Also, to save disk access time, I wanted to be able to re-activate a disconnected TSR program provided it was still in memory.

To accomplish this, I give each TSR program both connect and disconnect code. It also keeps a flag of data to tell whether it is activated or not. The first code in the program checks the flag to see whether it is connected or not. If it is not connected, then it installs itself. If it IS connected, then it de-installs itself. In both cases, it changes the flag to reflect its new status.

The connect routine saves any old vectors which this program will connect to, resets MEMTOP to protect itself, and installs its own vectors where appropriate. For example, Gnomus connects itself to WHO. It also marks the flag "installed".

The disconnect routine restore any old vectors it may have been connect to, resets MEMTOP to point past itself, and marks the

flag "uninstalled". Since I'm running out of room, I'll explain just how this works in conjunction with Exec's standard commands in the next issue. (TO BE CONTINUED)

---

### Advertising

Commercial advertising rates are \$50 for a full page, \$25 for a half page, and \$15 for a quarter page. Anything smaller is \$3.00 per column inch. A column is 3-3/4 inches wide by 10 inches tall. A full page is 7-5/8 inches wide. Noncommercial ads by subscribers are free.

---

PolyMorphic 8813 needs home. Make offer. Conway Spitler, P. O. Box 385, Fillmore, CA 93016-0385

---

Entire PolyMorphic System User Manual, System 88 User's Manual with Exec/96 addendum, & System 88 Operation Essentials On IBM disk. Al Levy, 516-293-8358

---

FOR SALE: Poly 8810 box with power supply and mother board. \$50 plus shipping. Charles A. Thompson, 2909 Rosedale Avenue, Dallas, Texas 75205-1532, (214)-368-8223.

---

### DISKS - MODEMS - PROMS - SOFTWARE - SPELL \*\*\* New Low Prices \*\*\*

1. MAXELL diskettes: 5-1/4" 10 hard sector -- \$10 per box.
2. Used diskettes: 5-1/4" 10 hard sector -- \$0.50 each.
3. Hayes Micromodem 100 (300 baud S-100 internal modem) \$20.  
(If you don't have a modem this is a cheap way to go.)
4. HayesSys modem software (for the Micromodem 100) \$10.
5. Abstract Systems Exec (Enhancements & bugs corrected) \$30.
6. Abstract Systems Proms (Enhancements & bugs corrected) \$35.
7. PolyGlot Library Volumes: \$6 each; 5 or more - \$5 each.
8. Hayes Smartmodem 1200B (IBM compatible internal) \$40.  
Abstract Systems, etc., 191 White Oaks Road,  
Williamstown, MA 01267, Phone: (413) 458-3597  
(Send \$1.00 for a complete catalog--[free with any order].)  
(Make check or money order payable to Ralph Kenyon.)

---

### Bit Bucket

I ran out of "bits of wisdom", so thought I'd run with a bit of "horse sense". Ever wonder where the expression "getting the bit in his teeth" comes from and why it means running away with

something? The "bit" is the cross bar which is placed in a horse's mouth and is connected to the reins. The cross bar normally fits in back of the front teeth and presses against the tender parts of the mouth. Pulling on the reins puts pressure on these tender parts and a well trained horse will turn without needing enough pressure to cause pain. But, an under trained rider may pull on the reins enough to cause the horse pain. Also, an under-trained horse will often not respond without a strong enough pull to cause pain.

Horses may sometimes be able to work the bit around in their mouth until they gets it in their teeth. Pull as much as you like, the horse with the bit in his teeth can go where he likes and is, for practical purposes, out of control. "Getting the bit in his teeth" has come to be used to describe the situation when someone takes charge and runs out of control.

**In This Issue**

Editorial . . . . .	1
Letters . . . . .	2
slist.GO. . . . .	2
PACK-CRASH . . . . .	3
The Other Guys . . . . .	6
PC Outline . . . . .	6
Managing TSR programs . . . . .	7
Advertisements . . . . .	9
The Bit Bucket . . . . .	9

**Coming Soon**

Conclusion of Managing TSR programs, Poly Meta, More: BASIC for Beginners, PC stuff, System Programmers Notes, Help, BugNotes, Public Domain Software, etc. PolyLetter: The First 10 Years.

**PolyLetter**  
 191 White Oaks Road  
 Williamstown, MA 01267  
 (413) 458-3597

Address Correction Requested

**FIRST CLASS MAIL**



Ralph E. Kenyon, Jr.      EXP:99#9  
 Abstract Systems, etc.      184  
 191 White Oaks Road  
 Williamstown, MA      01267-2256

© Copyright 1989 by Ralph E. Kenyon, Jr.

PolyLetter Editor and Publisher: Ralph Kenyon. Subscriptions: US \$18.00 yr., Canada \$20.00 yr., Overseas \$25.00 yr., payable in US dollars to Ralph Kenyon. Editorial Contributions: Your contributions to this newsletter are always welcome. Articles, suggestions, for articles, or questions you'd like answered are readily accepted. This is your newsletter; please help support it. Non-commercial subscriber ads are free of charge. PolyLetter is not affiliated with PolyMorphic Systems.

Back volumes of *PolyLetter* (1980 thru 1988) are available at reduced prices payable in US dollars to Ralph Kenyon. 1 - \$15, 2 - \$28, 3 - \$40, 4 - \$50, 5 - \$59, 6 - \$67, 7 - \$75; Canada add \$3 shipping, Overseas add \$10. Individual back issues are also available (\$3.50, \$4.00, \$5.00).



# PolyLetter



PolyLetter 90/4

Page 1

JUL/AUG 1990

## Editorial

It just dawned on me that PolyLetter has published over 10 years of issues. For some reason I was thinking we had another year to go to complete a full decade -- not so; we're there already. PolyLetter was started in 1980 by Mark Sutherland, then of Atlanta, GA, 4 years after PolyMorphic Systems entered the market with a full fledged computer.

The very first publication of PolyLetter actually was a one-page letter individually addressed to selected users, and including a two page enclosure; it was mailed on February 6, 1980. The first "proper" issue of PolyLetter included 6 pages of programs, hints, program reviews, announcements, a list of free programs from Poly, and advertisements. The Parrot on the Masthead, now known as "Mr. Squock", didn't appear until the next issue. Speaking of Birds, we still need a name for the ferocious one on the cover. Send in your suggestions.

Well, I've gone through the first 60 issues of PolyLetter and prepared a comprehensive index to the first decade. It's too big to print in one issue, so I'll publish it in parts. But, for those of you who would like to get the rest of it faster, as well as a fast way to find things in it, I have put the entire list in the next PolyGlott library volume. See "In the Public Domain" elsewhere in this issue.

## Letters

Dear Ralph,

September, 1990

How's this for a product announcement: NEW POLYMORPHIC EMULATOR RUNS AT TEN TIMES THE SPEED OF A REAL POLY.

Really? Well, not yet. But it could now be done. I was reading a Zilog data book yesterday and discovered that they now

build a 20 MHz Z-80 processor, which would be roughly 10 times as fast as a Poly. To take advantage of this, of course, one would have to build a CPU board to hold that Z-80, along with some very fast memory.

Ah, dreams. That project is probably never going to be built due to lack of demand in the Poly world. But it now COULD be built. How would you like to see your BASIC programs run 10 times faster? How quick can your quicksort really run? Anyone really want to know the answers to these questions? Probably not.

Other news: I recently left Telecorp to go back into the hardware design field; I'm now with a company called Scientific Games. They're based in Atlanta and they operate the equipment for the state lotteries in several states. My job (at least initially) will be designing the point-of-sale terminals that print and verify the lottery tickets. I may eventually be doing some of the firmware programming for these terminals too, and if so, the job will be much like the one I had at Chromatics. A little hardware, a little software. My new number at work is 404/984-7810; home is still 498-3556. -- Bob Bybee, Stone Mountain, GA.

[Ah dreams, yes. See "The New System" in PL 86/4 -- Ed.]

PolyLetter

August 7, 1990

Keep it up! Poly still runs once a week. I'd like to see more answers to PC questions -- PC editors WordStar, QEDIT, DBASE, PC files, Clarion, Venture, Corel. -- Russ Nobbs, Spokane WA.

## Managing TSR Programs

Continued from PL 90/3

My TSR memory management protocol has the connect and disconnect code as the very first code in the program. By having the

connect and disconnect code as the first code in the program, we can take advantage of some peculiarities of the 8080 and Poly's Exec. For the 8080 "00" is a machine language NOP (No OPeration). The 8080 does nothing and moves up to the next byte. For Exec, the "ZAP" command clears out USER memory all the way up to MEMTOP by making every byte "00". Exec also has a "START" command which tells the system to execute the program in USER memory. If we START the program when there is nothing there, the 8080 will race through all those 00 bytes and eventually get up to the protected program which is past MEMTOP. Since the ZAP command only affects USER memory, the lowest program above MEMTOP will be still in memory, and the 8080 will begin executing it. Of course, Exec is smarter than that. It keeps track of when a program is loaded. Under ordinary circumstances, if you try to START after you have ZAPPED, Exec will say "Nothing to run!". Of course, Exec also knows that human's are smarter, so has a way for us to override this limit. If we ENABLE the system, Exec will figure that a smart programmer knows what he or she is doing when he or she tells Exec to START. In that case, Exec will go ahead and START.

So, I write all my TSR programs so they can be disconnected from Exec with the ENABLE, ZAP, START, command sequence. (Some of them may be immediately re-installed by "START"; edit is an example.) By using ZAP and START, the lowest program in memory will be executed first. This insures that TSR programs are removed from the "high memory stack" in the reverse order that they are installed. Here is the generic memory management portion of my TSR programs which may be restarted. Note: this is for the body of the program and does not include the loader.

```
REFS SYSTEM.SY
REF MEMTOP
REF Msg
```

```
RELOC
```

```
Begin EQU 9-1 ;MEMTOP when installed
Start LXI H,Status
MOV A,M
ORA A
JNZ Discon
Install MVI M,OFFH ;FF = connected
LXI H,Begin ;Protect
```

```
SHLD MEMTOP ;our code
LXI H,con ;Tell 'em
CALL Message ;we're connected
;Other connect code goes here
RET

Discon MVI M,0 ;00 = disconnected
LXI H,End ;Clear the
SHLD MEMTOP ;protection
LXI H,dis ;Tell 'em
CALL Message ;we're disconnected
;Other disconnect code goes here
RET

;The main program code goes here

Message PUSH H
LXI H,Prog
CALL Msg
POP H
JMP Msg

Prog DB 'TSR program ',0
dis DB 'dis'
con DB 'connected',13,0

Status DS 1 ;Installed = FF, Not installed = 00
End EQU 9-1 ;MEMTOP when not installed

END
```

Now, some programs may be such that they mustn't be restarted. Some self modifying programs are an example. Others may change MEMTOP by grabbing big buffers, or have very extensive initialization code which we don't want sitting around just consuming memory. In this case, I write code which allows the program to be disconnected, but does not allow it to be reconnected. Here is the generic memory management portion of my TSR programs which may be restarted. Note: this is for the body of the program and does not include the loader.

```
REFS SYSTEM.SY
REF MEMTOP
REF Msg
REF Warn
```

```
RELOC
```

```
Begin EQU 9-1 ;MEMTOP when installed
Install LXI H,Status
MVI M,OFFH ;FF = connected
LXI H,Discon ;Protect
SHLD MEMTOP ;our code
LXI H,con ;Tell 'em
CALL Message ;we're connected
;Other connect code goes here
RET

Discon LXI H,Status
```



```

MOV A,M
ORA A
JZ Warn ;Already disconnected!
MVI M,0 ;00 = disconnected
LXI H,End ;Clear the
SHLD MEMTOP ;protection
LXI H,dis ;Tell 'em
CALL Message ;we're disconnected
;Other disconnect code goes here
RET

```

```

;The main program code goes here

```

```

Message PUSH H
LXI H,Prog
CALL Msg
POP H
JMP Msg

```

```

Prog DB 'TSR program ',0
dis DB 'dis'
con DB 'connected',13,0

```

```

Status DS 1 ;Installed = FF, Not installed = 00
End EQU $-1 ;MEMTOP when not installed

```

```

END

```

When these programs are filled out with appropriate code, Asmb.GO will produce a relocatable file. (You must specify that the output file is to have extension ".RL" because Poly never made Asmb that smart.) Once you have a relocatable file, you can convert it to a self relocating program using MakeRel, or you can load it directly with LoadRel.GO. (Exec/(A;S) knows how to load and execute a .RL file.)

With this memory management strategy, any number of TSR programs may be "stacked up" in high memory and be unstacked in reverse order. Of course, certain precautions must be taken in writing the TSR program to make sure it only uses its own memory -- but that's another story for another time.

### Basic Pointer

Want to make sure the user gives you a decimal fraction on input? Here's how you can get the "point" across. When you get to the point where you want to input the number, put the decimal point into the keyboard buffer first. Then, when BASIC stops to get input from the user, it will get the point first. Here's a simple example program

```

10 ON ERROR PRINT "Input error." \Z=CALL(0,")
20 OUT 0,CHR$(24)+".\" \REM CTRL-X to clear buffer first
30 INPUT "Give me the decimal part ",A
40 PRINT "Your number is" \M GOTO 20

```

Thanks to Charles Steinhauser for posing this problem. (Of course, some users could make the whole effort pointless by hitting the DELETE key first.)

### Serial vs. Parallel

by Bob Bybee

Some of PolyLetter's readers have asked me about serial and parallel printers. What's the difference? Why has the industry moved away from serial-interface printers? Which is better? Let's look at some of these issues.

First, how does a serial interface differ from a parallel one? On an IBM-PC, both use a 25-pin connector, so you'd think they would be comparable at least in the number of wires used. But no, the 25-pin connector was chosen by IBM just because it was readily available and would fit on the rear of the machine. Neither the serial nor the parallel interface actually uses all 25 pins. (On some IBM AT machines, the serial port is on a 9-pin connector!)

So let's look at how serial and parallel interfaces actually function. We have to start by understanding how data is sent to a printer. Data is sent in bytes, and a byte is 8 bits. (You've heard this one before.) But what's a bit? Hmm... let's not worry about it too much. Think of it as a rock. Just remember we have to throw 8 rocks, from somewhere to somewhere else, before a character gets printed. To print the name POLY, which has 4 characters, we'd have to throw  $8 \times 4 = 32$  rocks.

There are two types of bits... oops, I mean rocks. Let's think of them as white rocks and black rocks. By sending them in different combinations, we can make different characters print on the printer. The order in which we throw the rocks is important too: white-black is different from black-white, as far as the printer is concerned.

In a serial interface, we have one man throwing rocks. To make a character print, he has to throw 8 rocks, as we said. But in a parallel interface, we have 8 men all throwing their rocks at the same time. All else being equal, it's going to be faster for 8 men to throw a given number of rocks, than for one man to throw them all. So in theory, a parallel interface can be faster than a serial one.

But the "men" are actually wires in a cable, and it's going to take 8 wires for a parallel interface, and only one for a serial interface. So the cable we need for

parallel is going to be more expensive. That becomes important only for very long cables.

The earliest printers had serial interfaces. Why? Because they weren't usually just printers. They were printing TERMINALS, with a keyboard too. And one of those, with a modem, was what you needed to dial into a mainframe computer, and that was the only way most of us could get access to a computer at all, 15 years ago. When we started getting our own small computers, the easiest way to get a printer was to buy one of these terminals. Some companies built terminals with and without a keyboard, and you could save a little money buying the keyboard-less version when possible. But a serial interface was just about standard equipment.

One company that bucked this trend was Centronics. They built printers with parallel interfaces, for connecting to minicomputers and mainframes. The term "Centronics interface" became just about synonymous with "parallel interface." When IBM introduced their PC, they provided it with a parallel interface that could plug into a Centronics printer.

But IBM always has to do something squirrely. The standard connector on a Centronics printer was a 36-pin job. IBM decided they didn't like that (or it didn't fit well on the rear of the PC). So they outfitted their parallel interface with a 25-pin "D"-type connector, just like all of the serial ports in the world. They also designed a special cable that had their 25-pin connector at one end, and a Centronics connector at the other end.

From then on, just about every printer in the world has used a parallel interface. It would be suicide to sell a printer today that didn't have an IBM-PC compatible interface on it. (Unless you're Apple... that's another story.) And as it turns out, it's a little cheaper to build a parallel interface than it is to build a serial one.

There's also fewer hassles with hooking up a parallel printer. Plug it in. That's all. With a serial printer, as many Poly owners know, you have to worry about how the serial port cable is wired, what's the baud rate, how does the handshaking work, and so on. On the Poly, there's also the

problem of that little "serial header" plug inside the machine, which can be wired in a variety of ways.

If you want to connect a parallel printer to your Poly, you have several choices. Some printers also have a serial port installed as standard equipment; if so, just use it! That's the easy way out. But not too many printers have this. Some offer a serial interface as an option, usually costing \$50 and up.

The other way is to buy a serial-to-parallel converter box, which can take the data from your Poly's serial port and convert it into what a parallel printer needs. You can get a simple one for about \$50. I've interfaced one of these cheap ones to a Poly and a Panasonic KXP-1180 printer (which cost about \$180 and provides darn good quality for a 9-pin printer). That's probably the least expensive way to put a new printer on an old Poly these days.

If you want to spend a little more, you can get a "print buffer" box which has memory in it, and "absorbs" data coming from the Poly as fast as the Poly can send it. Then it slowly feeds that data out to the printer, so you can start using your Poly again while the printer prints. These print buffers are primarily designed to free up your computer while your printer runs, but they often have the additional feature of giving you a serial input port and a parallel output. These start at around \$200.

If anyone needs help with this kind of interfacing project, let me know. [Bob can be reached at Poly Peripherals, 404/498-3556. Note: Al Levy offered such a buffer for sale in PL 89/3. Al says he still has one or two around.]

### **Announcements**

Tussey Computer Products has changed its name to Swan Technologies. Swan is located at 3075 Research Drive, State College, PA 16801. I recently heard a report on national TV which identified Swan Technologies as providing the best customer satisfaction on repairs for a mail order firm. The name change was because more people had heard about their Swan line of computers than the old company name itself. Swan announced that all systems are now



available in tower as well as desktop models, and price reductions on 386 and 486 systems. Contact Chris Perantoni at (814) 238-1820.

### *In the Public Domain*

PGL-V-33 contains a complete index of the first 10 years of PolyLetter, as well as a complete list of all programs in the PolyGlot Library. Included is {A;S} Search, a fast search program which will search for any text string you want. It is not case-sensitive, so will find all combinations of upper and lower case in the file being searched.

Disk PGL-V-33 has 5 files on it, 55 free entries. 350 sectors in use, 0 sectors deleted, 0 sectors free.

Size	Addr	La	Sa	flags	Name.
4	7	101	101		HF.DX
3	B	3200	3200	N	Search.GO
259	E	0	0	N	POLYLETTER-INDEX.TX
74	111	0	0	N	POLYGLOT-LIBRARY-INDEX.TX
1	15D	2C00	2C00	N	HELP.GO

### *PolyLetter - the First 10 Years (part 1)*

*# Requires Drive BugNote 006.0	Ralph Kenyon	86/5/08
# HELP	Ralph Kenyon	86/2/06
16 by 64 display	Mark Sutherland	80/4/02
16K Memory board bug	Mark Sutherland	80/4/02
16K Memory Card Update	Ralph Kenyon	86/6/03
1983 Author Index	PolyLetter	84/1/06
1983 Index	PolyLetter	84/1/06
1983 Subject Index	PolyLetter	84/1/06
1984 Author Index	PolyLetter	85/1/05
1984 Index	PolyLetter	85/1/05
1984 Resale Value of a Poly	Russ Nobbs	86/3/04
1984 Subject Index	PolyLetter	85/1/05
80386 S-100 boards by Lomas	Al Levy	87/1/05
8048 Cross Assembler		87/1/08
8048 Cross assembler comments	Ron Moffatt	87/4/04
8080 Architecture	Ralph Kenyon	88/3/04
8080 Assembly codes	Ralph Kenyon	88/3/05
8080 Emulation History	Bob Bybee	89/5/06
88-MS	Mark Sutherland	80/1/04
88/HD	PolyMorphic Systems	81/4/02
96 tpi PROMS	Ralph Kenyon	84/5/02
? HELP	Ralph Kenyon	86/2/06
A Good Printer for Polys	PolyLetter	82/4/05
A Note from Poly	Gordon Furman	80/2/01
A Two Drive-8810 Modifying the ROMS	Bob Bybee	84/3/02
A Two-Drive 8810: Mechanical	Bob Bybee	84/2/04
A View from Comdex	PolyLetter	83/3/06
Abort.GO program description	PolyGlot	89/2/06
About PolyLetter	Mark Sutherland	80/3/01
About Public Domain Software	Ralph Kenyon	86/2/05
ABSTRACT SYSTEMS POLICY	Ralph Kenyon	87/6/05
ACCURATE-DIVIDER.BS Program Description		86/5/09
ACT (<A><C><T>) Report	Al Levy	87/1/05
Adding a Solid State Disk to Poly	John J. Warkentin	85/2/03
Adding a Solid State Disk to Poly	John J. Warkentin	85/4/03
Adding a Solid State Disk to Poly	John J. Warkentin	85/5/06
Adding Mass Storage to Polys	PolyLetter	82/6/03
Address List: PolyLetter Subscriber	Charles Steinhauser	86/1/04
Adventure Maze Map	Ralph Kenyon	86/2/07
Adventure Patches	Ralph Kenyon	85/5/03
ALTAIR	Bob Bybee	81/5/03
Anatomy of a Bad Memory Chip	Ralph Kenyon	86/4/04
And the Bit Goes On (8 bit serial)	Don Barrett	84/3/04
And the torch has passed	Charles Steinhauser	85/2/01
And the Torch is Passed	Bob Bybee	83/6/01
APR 80 Disk of Month listing		80/2/06
APR 80 Disk of Month Listing	PolyLetter	84/1/06
ARCSIN bug	Ralph Kenyon	88/2/09
ARISE.GO	Ralph Kenyon	87/1/04
ARTIL.BS program description	Bob Bybee	81/6/05
ASCII CHART	Mike Stanford	80/5/13
ASCII table	Ralph Kenyon	88/3/06
ASIN Bug	Ralph Kenyon	83/2/07
Asmb Tutorial	Ralph Kenyon	81/3/08
Asmb.GO Housekeeping	Ralph Kenyon	87/3/06
Assembly code in BASIC string	Don Moe	80/4/07
Assembly codes for the 8080	Ralph Kenyon	88/3/05
Assembly Language Input - WHO	Ralph Kenyon	86/4/06
Assembly Language Interface: BASIC	Ralph Kenyon	86/5/07
Assembly Language OP-CODES	PolyLetter	81/3/02
Assembly Language Output - WH1	Ralph Kenyon	86/4/06
Assembly Language Program	Ralph Kenyon	86/4/06
Assembly Language References in PL	Ralph Kenyon	86/6/03
Assembly Language Tutorial	Ralph Kenyon	80/4/09
Assembly Language Tutorial	Ralph Kenyon	81/1/06
Assembly Source for Sio.PS	Ralph Kenyon	88/1/05
AUG 80 Disk of Month listing		80/4/08
AUG 80 Disk of Month Listing	PolyLetter	84/1/06
Automatic Linefeed Disable	Bob Jones	87/1/04
Back Issues of PolyLetter, Prices		87/3/06
BACKGAMMON.BS program description	Bob Bybee	81/6/05
Backslash and FORMAT	Al Levy	86/3/01
Backspace and FORMAT	Al Levy	86/3/02
BACKUP.GO program description	Chuck Thompson	81/6/04
Bad Memory Chip: Anatomy of	Ralph Kenyon	86/4/04
BAGELS.BS Program Description		87/4/07
BASE.BS Program Description		87/5/08
BASES.GO description	PolyLetter	83/2/06
BASIC ASIN Bug	Ralph Kenyon	83/2/07
BASIC Assembly Language Interface	Ralph Kenyon	86/5/07
BASIC bug in ARCSIN	Ralph Kenyon	88/2/09
BASIC Bug STOP	PolyLetter	82/1/05
BASIC C02 Enhancements	PolyLetter	81/1/01
BASIC C04	Chuck Thompson	83/4/08
BASIC CALLs to system	Don Moe	80/4/07
BASIC COS Bug	Ralph Kenyon	83/3/07
BASIC COS bug	Ralph Kenyon	88/3/08
BASIC Data Types	Ralph Kenyon	86/6/05
BASIC DEF FN Eliminate Cursor	Mark Sutherland	80/1/04
BASIC Error processing	John Warkentin	81/2/03
BASIC FILE bugs	PolyLetter	81/2/03
BASIC Function Draw a Border	Al Levy	86/1/07

BASIC function MOD	Russ Hobbs	81/4/04	Bug in BASIC SIN	Ralph Kenyon	83/3/07
BASIC Function To Skip Lines	Charles Thompson	86/1/06	Bug in BASIC WRITE	Ralph Kenyon	83/1/03
BASIC Function To Strip Blanks	Charles Thompson	86/1/06	Bug in boot	Ralph Kenyon	83/1/03
BASIC function Yes or No	Jonathan Wolfret	81/3/06	Bug in Edit Exit	Ralph Kenyon	83/3/07
BASIC Function: Recursion	Ralph Kenyon	86/2/06	Bug in Edit Input	Ralph Kenyon	83/3/07
BASIC INPUT Bug	Ralph Kenyon	83/1/03	Bug in FTP	Ralph Kenyon	83/3/07
BASIC Input bug BugNote 001.1	Ralph Kenyon	86/2/06	Bug in Gfid #	Ralph Kenyon	83/3/07
BASIC LEN input: BugNote 008.0	Ralph Kenyon	86/6/07	Bug in Gfid Enter	Ralph Kenyon	83/3/07
BASIC LOAD times Comparison	John Warkentin	81/2/08	Bug in OM ERROR	PolyLetter	83/2/07
BASIC Machine Language Calls		80/3/03	Bug in PACK	Ralph Kenyon	83/1/03
BASIC Program COPYLINE.BS	Al Levy	86/1/02	Bug in ROM 81	Ralph Kenyon	83/1/03
BASIC Program HANOI-G.BS	Ralph Kenyon	86/5/08	Bug in Sio Input	Ralph Kenyon	83/3/07
BASIC Program HANOI.BS	Ralph Kenyon	86/2/06	BUG Index HELP	Abstract Systems	89/2/09
BASIC Program Maintenance Hints	Ralph Kenyon	86/5/08	BUG OUT - BugNote 24	Ralph Kenyon	88/1/09
BASIC Programming	Ralph Kenyon	86/6/05	Bug: The First	Ralph Kenyon	86/5/05
BASIC Programs: Fast Loading	Ralph Kenyon	86/6/10	BugNote 001.1 BASIC Input	Ralph Kenyon	86/2/06
BASIC READ Bug	Ralph Kenyon	83/1/03	BugNote 002.0 RENAME with "*" bug	Ralph Kenyon	86/3/06
BASIC Review	PolyLetter	85/2/05	BugNote 003.0 Double Error Message	Ralph Kenyon	86/4/08
BASIC SHORTHAND	Frank Stearns	81/1/04	BugNote 004.0 PROMS Dio Read Error	Ralph Kenyon	86/4/08
BASIC SIN Bug	Ralph Kenyon	83/3/07	BugNote 005.0 Pack Syntax Failure	Ralph Kenyon	86/5/08
BASIC SIN bug	Ralph Kenyon	88/3/08	BugNote 006.0 "*" Requires Drive	Ralph Kenyon	86/5/08
BASIC Software Maintenance Hints	Ralph Kenyon	86/5/08	BugNote 007.0 MS Controller Ports	Ralph Kenyon	86/5/08
BASIC String initializing	Ralph Kenyon	88/1/10	BugNote 008.0 BASIC LEN input	Ralph Kenyon	86/6/07
BASIC system calls	Bob Bybee	80/5/04	BugNote 009.0 Pack Error: NO Abort	Ralph Kenyon	86/6/07
BASIC Time Clock	Dan DeForest	80/3/08	BugNote 010.0 Edit exit bug	Ralph Kenyon	87/1/07
BASIC User Defined Functions	Charles Thompson	82/1/08	BugNote 011.0 RDB exit bug	Ralph Kenyon	87/3/08
BASIC User Defined Functions	Charles Thompson	86/1/06	BugNote 012.0 boot bug	Ralph Kenyon	87/4/10
BASIC Variables	Ralph Kenyon	86/6/06	BugNote 018.0 - FTP	Abstract Systems	89/3/09
BASIC WRITE Bug	Ralph Kenyon	83/1/03	BugNote 033.0 - DIGITS	Abstract Systems	89/3/05
BASIC-USER-FUNCTION-TABS.BS Program Description		86/5/09	BugNote 039.1 - EDIT-5	Abstract Systems	89/5/09
BASIC-USER-FUNCTIONS.BS Program Description		86/5/09	BugNote 13 - ARCSIN	Ralph Kenyon	88/2/09
BATTLESHIP.BS description	Bob Bybee	81/3/05	BugNote 15 - SIN/COS	Ralph Kenyon	88/3/08
Beginners, For	Abstract Systems	89/6/09	BugNote 16 EDIT-2	Ralph Kenyon	88/3/08
Big Book of Photo Copier Humor Rev	PolyLetter	85/2/03	BugNote 17.0 Sio.PS	Abstract Systems	89/1/09
Binary Device	Ralph Kenyon	88/3/10	BugNote 24 - OUT	Ralph Kenyon	88/1/09
BIORHYTHM-1.BS Program Description		86/5/09	BugNote SAVE on Exec/83	Mark Sutherland	80/3/02
BIORHYTHM-2.BS Program Description		86/5/09	BUGS HELP	Abstract Systems	89/2/09
BIORHYTHM.BS description	Bob Bybee	81/3/05	Bugs in BASIC C03	Ralph Kenyon	82/1/06
Bit Bucket	PolyLetter	85/5/01	Bugs in Exec/93	PolyLetter	81/2/03
Bit Bucket	PolyLetter	85/6/07	BUGS!	Ralph Kenyon	83/1/03
Blinking Load Light	PolyLetter	82/4/02	BusinessVision II	PolyLetter	89/3/01
Bob Bybee's New Address	Bob Bybee	86/3/02	Buying Poly's Software Source	Bob Bybee	86/3/02
Bomb Shelter	Mark Sutherland	80/3/06	Bypassing FORMAT.IN	Ralph Kenyon	86/5/10
Bomb Shelters	Russ Hobbs	81/4/08	Bypassing the printer driver	Chuck Thompson	81/3/05
Book Review	Charles Gross	81/5/03	C, an introduction	Bob Bybee	88/3/01
boot Bug	Ralph Kenyon	83/1/03	C, More	Ralph Kenyon	88/4/01
boot HELP	Charles Thompson	80/4/04	Cache.ZO - Explained	Joe Spahn	82/6/02
boot HELP	G. R. Gamble	80/2/05	CALL(0,"string")	PolyLetter	81/1/02
Bove's Theorem	Murphy	88/6/06	CALL(1)	PolyLetter	81/1/02
BOWLING.BS description	PolyLetter	82/4/06	CALLing Overlays from BASIC	Chuck Thompson	82/3/08
BOXING.BS Program Description		87/4/08	CALLS to system from BASIC	Don Moe	80/4/07
Brooks' Law	Murphy	88/6/06	Canada Bill Jones' Motto	Murphy	88/6/06
BUG DIGITS	Abstract Systems	89/3/05	Cann's Axiom	Murphy	88/6/06
BUG EDIT HELP	Abstract Systems	89/5/09	CANNON.BS Program Description		87/4/07
Bug in 16K Memory boards	Mark Sutherland	80/4/02	CASHFLOW Wanted	Ralph Kenyon	86/6/02
Bug in BASIC - Stop	PolyLetter	82/1/05	CASHFLOW.BS description		87/2/09
Bug in BASIC ASIN	Ralph Kenyon	83/2/07	Cassette use	Chuck Thompson	87/5/01
Bug in BASIC COS	Ralph Kenyon	83/3/07	CBUF	Bob Bybee	81/1/05
Bug in BASIC INPUT	Ralph Kenyon	83/1/03	CHANGE.GO description	PolyLetter	82/4/06
Bug in BASIC READ	Ralph Kenyon	83/1/03	CHASE.BS Program Description		87/4/07



Cheap Printers	Bob Bybee	86/4/03	CP/M Corner	PolyLetter	82/5/03
Checkbook Program Wanted	Charles Mach	86/6/01	CP/M Corner	PolyLetter	83/2/03
Checkbook system for IBM	Ralph Kenyon	89/1/07	CP/M Corner	PolyLetter	83/3/04
Chemistry program descriptions		87/2/08	CP/M Corner	PolyLetter	83/4/03
CHESS.GO description	Bob Bybee	81/3/05	CP/M Corner	PolyLetter	83/5/04
Chess.GO Fix	PolyLetter	82/6/04	CP/M Corner	Russ Nobbs	82/4/03
Chip Seating	PolyLetter	84/2/03	CP/M for the Poly	Mark Sutherland	80/4/01
Christmass Tree	Ralph Kenyon	88/6/10	CP/M from Poly	PolyMorphic Systems	81/4/01
Clarke's Third Law	Murphy	88/6/06	CP/M is Now Available	Bob Bybee	80/5/01
CLEAN HELP	Ralph Kenyon	88/3/09	CP/M Program Exchanges	Russ Nobbs	82/6/05
CLEAN-TAIL.BS program		87/2/10	Crash Recover	Ralph Kenyon	88/2/05
Cleaning boards	Jim Trahan	87/3/02	CRASH!	Ralph Kenyon	87/4/05
Cleaning Disk Drives	Ralph Kenyon	86/6/02	Cross assembler comments	Ron Moffatt	87/4/04
Cleaning Edge Connector "Fingers"	PolyLetter	84/2/04	Cursor Control in Edit	Ralph Kenyon	87/2/02
Cleaning Keyboard Keys	Ralph Kenyon	86/6/02	DATA-ENTRY.BS description	PolyLetter	83/2/06
Cleaning MS drives	PolyMorphic Systems	81/4/03	DataDisk, CopySelf program ad	Ralph Kenyon	81/4/05
CLEARNEW.GO program description	Chuck Thompson	81/6/04	Deadline-Dan's Demo Demonstration	Murphy	88/6/06
Clock in BASIC	Dan DeForest	80/3/08	Deadline-Dan's Demon	Murphy	88/6/06
Clock Program Modified	Sam Black	80/4/04	Dealer Written Software	PolyMorphic Systems	81/4/07
CMDA	Bob Bybee	81/1/05	DEC 81 Disk of Month Listing	PolyLetter	84/1/05
CMDD	Bob Bybee	81/1/05	DECHEXAMAL	Mike Stanford	80/5/04
CMDF	Bob Bybee	81/1/05	DELETE HELP	Ralph Kenyon	88/3/09
CMDF.AS Program Description		86/4/09	Demian's Observation	Murphy	88/6/06
CMDN	Bob Bybee	81/1/05	Description of PGL-V-26	Ralph Kenyon	88/1/07
CMDP	Bob Bybee	81/1/05	Description of PGL-V-27	Ralph Kenyon	88/2/06
Cold Keyboard	Bob Bybee	82/1/07	Description of PGL-V-28	Ralph Kenyon	88/3/06
Command File Internals	Bob Bybee	81/1/05	Details from Dallas	Chuck Thompson	81/3/04
Command files	PolyLetter	84/1/03	Device Driver Direct Access	Ralph Kenyon	86/5/06
Command Printer HELP	Abstract Systems	89/1/09	Diablo 620 Setup	Chuck Thompson	87/5/01
COMMAND UNDELETE HELP	Abstract Systems	89/6/09	Diablo Habits	Michael Aquino	82/5/04
Commenting ESCAPE Libraries in Edit	Ralph Kenyon	89/4/03	Diagnostics	Bob Bybee	84/3/03
Communicating at 9600 Baud	PolyLetter	85/1/02	dial.GO program description	Ralph Kenyon	89/3/09
COMP-DISK.GO program description	Chuck Thompson	81/6/05	Dictionary for the Poly	ISS	81/5/03
COMPARE PROGRAM HELP	Abstract Systems	89/4/10	Digital Research Board Modification	Joseph Toman	86/5/02
COMPARE/TWO.BS Program Description		87/5/08	Digital Research Information	Digital Research	81/5/02
Comparing BASIC LOAD times	John Warkentin	81/2/08	DIGITS	Ralph Kenyon	89/3/03
Compile Small C Programs	Ron Cain	88/4/05	DIGITS BUG	Abstract Systems	89/3/05
COMPRESS.BS Program Description		87/5/08	DIGITS HELP	Abstract Systems	89/3/05
Compuserve	PolyLetter	82/1/02	DIGITS preservation program	Ralph Kenyon	89/3/04
Confessions of a Past Editor	Frank Stearns	86/5/03	Dio Read Error PROMS BugNote 004.0	Ralph Kenyon	86/4/08
Confidence Test failure	Ralph Kenyon	87/6/10	DIRCOPY PROGRAM HELP	Abstract Systems	89/6/09
Confidence Test Plug	Russ Nobbs	86/6/07	Direct keyboard input for FORMAT	Chuck Thompson	81/3/05
Confidence Testing the PolyMorphic	Bob Bybee	82/2/07	DIS80.GO description	PolyLetter	83/4/04
Connecting 2 88-MS's	Ralph Kenyon	89/1/01	Disable Automatic Linefeed	Bob Jones	87/1/04
Conversation With Poly, A	Bob Bybee	86/3/02	DisAsmb ad	Ralph Kenyon	81/5/05
Convert decimal to HEX	Mike Stanford	80/5/04	DisAsmb.BS program ad	Ralph Kenyon	81/4/05
Converting from SAVEF to SAVE	Ralph Kenyon	86/6/03	DisAsmb.BS Review	Mark Sutherland	80/2/03
COPY COMMAND HELP	Ralph Kenyon	87/4/08	Disk Directory Full	Ralph Kenyon	89/3/03
COPYLINE.BS: BASIC Program	Al Levy	86/1/02	Disk Drive Cleaning	Ralph Kenyon	86/6/02
CopySelf DataDisk program ad	Ralph Kenyon	81/4/05	Disk Drive Speed Setting	Ralph Kenyon	86/6/02
COS bug	Ralph Kenyon	88/3/08	Disk Failure Modes	PolyLetter	83/6/09
COS Bug in BASIC	Ralph Kenyon	83/3/07	Disk Holder, Magnetic	Mark Sutherland	80/2/05
COUNT.GO program description	Bob Bybee	81/4/05	Disk of Month APR-80 listing	PolyGlot	89/2/07
Cover Your Keyboard	Ralph Kenyon	89/1/05	Disk of Month AUG-80 listing	PolyGlot	89/2/07
CP/M	Al Levy	86/2/02	Disk of Month DEC-81 listing	PolyGlot	89/2/07
CP/M	Al Levy	86/2/04	Disk of Month JAN-81 listing	PolyGlot	89/2/07
CP/M	Ralph Kenyon	86/6/02	Disk of Month JUL-81 listing	PolyGlot	89/2/07
CP/M	Ralph Kenyon	87/2/05	Disk of Month JUL-82 listing	PolyGlot	89/2/07
CP/M Corner	Bob Bybee	81/5/02	Disk of Month JUL-83 listing	PolyGlot	89/2/07
CP/M Corner	Dick Jacobi	82/1/07	Disk of Month July 83 listing	PolyLetter	83/4/04

Disk of Month MAR-81 listing	PolyGlot	89/2/07	Editorial	Mark Sutherland	80/3/01
Disk of Month MAR-82 listing	PolyGlot	89/2/07	Editorial	PolyLetter	85/1/01
Disk of Month MAR-83 listing	PolyGlot	89/2/07	Editorial	PolyLetter	85/2/01
Disk of Month March 82 Listing	PolyLetter	82/2/05	Editorial	PolyLetter	85/3/01
Disk of Month MAY-81 listing	PolyGlot	89/2/07	Editorial	PolyLetter	85/6/01
Disk of Month NOV-84 listing	PolyGlot	89/2/07	Editorial	Ralph Kenyon	86/2/01
Disk of the month APR 80 listing		80/2/06	Editorial	Ralph Kenyon	86/3/01
Disk of the month AUG 80 listing		80/4/08	Editorial	Ralph Kenyon	86/4/01
Disk of the Month MAR 82 listing	PolyLetter	83/2/06	Editorial	Ralph Kenyon	86/5/01
Disk of the Month: APR-80 listing		86/3/07	Editorial	Ralph Kenyon	86/6/01
Disk of the Month: AUG-80 listing		86/3/07	Editorial	Ralph Kenyon	88/1/01
Disk of the Month: DEC-81 listing		86/3/07	Editorial	Ralph Kenyon	88/2/01
Disk of the Month: JAN-81 listing		86/3/07	Editorial	Ralph Kenyon	88/3/01
Disk of the Month: JUL-81 listing		86/3/07	Editorial	Ralph Kenyon	88/4/01
Disk of the Month: JUL-82 listing		86/3/07	Editorial	Ralph Kenyon	88/5/01
Disk of the Month: JUL-83 listing		86/3/07	Editorial	Ralph Kenyon	88/6/01
Disk of the Month: MAR-81 listing		86/3/07	Editorial	Ralph Kenyon	89/1/01
Disk of the Month: MAR-82 listing		86/3/07	Editorial	Ralph Kenyon	89/2/01
Disk of the Month: MAR-83 listing		86/3/07	Editorial	Ralph Kenyon	89/3/01
Disk of the Month: MAY-81 listing		86/3/07	Editorial	Ralph Kenyon	89/4/01
Disk of the Month: NOV-84 listing		86/3/07	Editorial	Ralph Kenyon	89/5/01
Disk-of-the-month JUL 81	Bob Bybee	81/4/05	Editorial	Ralph Kenyon	89/6/01
Disk-of-the-Month MAY-81	PolyLetter	81/3/05	Eliminating Automatic LineFeed	J. H. McNally	82/2/02
Disk-of-the-month NOV 81	Bob Bybee	81/6/05	Emulator Evaluation	Ralph Kenyon	89/5/08
Diskettes	Bob Bybee	83/3/07	Emulator for Poly on PC's	Bob Bybee	89/5/01
Display Time as Hours and Minutes	Al Levy	86/1/07	ENABLE COMMAND HELP	Ralph Kenyon	87/1/07
Documentation for Small C	Ron Cain	88/4/05	Epson RX80	Dr. Briarty	87/1/02
Don Moe departs	PolyLetter	81/3/03	Eratic Memory	Chris Bagley	87/1/02
Don's Bag of Tricks	Don Moe	88/3/04	Error messages for the MS	Russ Nobbs	81/4/06
Don't Clean Drive heads	Don Moe	81/2/02	Error processing in BASIC	John Warkentin	81/2/03
Donald Moe is Back	Don Moe	88/3/04	ERROR.GO description	PolyLetter	83/4/04
Donald Moe is Back - Again	Don Moe	80/4/07	ESC-Control-K	PolyLetter	84/3/01
Double Error Message BugNote 003.0	Ralph Kenyon	86/4/08	ESCAPE library comments	Ralph Kenyon	89/4/03
Double Spacing	Ralph Kenyon	80/3/08	Escaping the Front Panel	Russ Nobbs	81/4/08
DownTime	Ralph Kenyon	89/3/02	Estridge's Law	Murphy	88/6/06
DownTime -- EPROM	Ralph Kenyon	89/4/02	Ethics	Bob Bybee	87/6/03
Dr. Caligari's Come-back	Murphy	88/6/06	Ethics	Ralph Kenyon	87/6/04
Draw a Border: BASIC Function	Al Levy	86/1/07	EVAL explained	Charles Thompson	80/5/02
Drive Problem	Constantin Pavloff	86/6/02	EVAL.PM	Ralph Kenyon	89/5/08
Drive Service	ISR	89/6/01	Evaluating the Poly Emulator	Ralph Kenyon	89/5/08
DSDD Mass Storage	Mark Sutherland	80/1/04	Exchequer Review	Ralph Kenyon	89/1/07
DWP 230 Printer Hookup	Ralph Kenyon	87/5/03	Exec/83 SAVE bug	Mark Sutherland	80/3/02
DX.GO description	PolyLetter	82/4/06	Exec/96	Chuck Thompson	83/4/08
East Coast Service	Davis McCarn	87/3/01	Exec/(A)S)	Ralph Kenyon	87/3/04
edit	Ralph Kenyon	87/6/01	ExtraPOLYations	Ralph Kenyon	80/4/09
edit	Ralph Kenyon	88/5/07	ExtraPOLYations	Ralph Kenyon	81/1/06
Edit 4.1.0	PolyMorphic Systems	81/5/02	ExtraPolyations	Ralph Kenyon	81/3/08
Edit bug	Ralph Kenyon	88/3/08	Fast Loading BASIC Programs	Ralph Kenyon	86/6/10
EDIT BUG HELP	Abstract Systems	89/5/09	Fast Sorting	Bill Sullivan	81/3/06
Edit disaster recovery	Chuck Thompson	81/3/04	Favorite DOS programs	Al Levy	89/2/01
Edit Exit Bug	Ralph Kenyon	83/3/07	File Channel bug using MAT	Ralph Kenyon	82/1/06
Edit Input Bug	Ralph Kenyon	83/3/07	FILE:X,DEL	PolyLetter	81/1/01
Edit Secrets	Ralph Kenyon	89/4/03	Finagle's Rules	Murphy	88/6/06
edit.GO Program Description	Ralph Kenyon	87/5/01	Financial Programs Wanted	James Salinger	86/4/09
Editor Library functions	Russ Nobbs	81/3/06	FIND.GO description	PolyLetter	82/4/06
Editor's Chips	PolyLetter	84/2/01	First Bug	Ralph Kenyon	86/5/05
Editorial	Bob Bybee	81/1/01	Flaky 16K Memory Boards	Mark Sutherland	80/4/02
Editorial	Charles Steinhauser	86/1/01	FLIES.BS program description	Bob Bybee	81/4/05
Editorial	Mark Sutherland	80/1/01	flip HELP	Charles Thompson	80/4/04
Editorial	Mark Sutherland	80/2/01	FNTIMER.BS Description	PolyLetter	81/2/08



fold HELP	Charles Thompson	80/4/04	HELLER'S LAW	Murphy	88/6/07
Folding Strings in BASIC	Mark Sutherland	80/3/03	HELP #	Ralph Kenyon	86/2/06
FOOTBALL FORECASTING MODEL	Earl Gilbreath	87/2/03	HELP - A Listing of Help Files	Ralph Kenyon	86/4/07
Football Pointsread Analyzer	PolyLetter	89/3/02	HELP - How come?	Ralph Kenyon	89/3/03
For Adventure Fans	Ralph Kenyon	85/4/02	HELP ?	Ralph Kenyon	86/2/06
For Programmers Only	Mark Sutherland	80/1/03	HELP BASIC DIGITS	Abstract Systems	89/3/05
Form.OV review	Mark Sutherland	80/1/05	HELP boot	Charles Thompson	80/4/04
FORMAT 5.5 upgrade bug	Ralph Kenyon	82/1/06	HELP boot	G. R. Gamble	80/2/05
FORMAT and the Backslash	Al Levy	86/3/01	HELP BUG DIGITS	Abstract Systems	89/3/05
FORMAT and the Backspace	Al Levy	86/3/02	HELP BUG EDIT-5	Abstract Systems	89/5/09
FORMAT direct keyboard input	Chuck Thompson	81/3/05	HELP BUG Index	Abstract Systems	89/2/09
FORMAT.GO; how it works	Ralph Kenyon	83/4/06	HELP BUGS	Abstract Systems	89/2/09
FORTH	Ralph Kenyon	81/5/07	HELP COMMAND COPY	Ralph Kenyon	86/5/06
FORTH - PGL-V-29	PolyGlot	89/1/09	HELP COMMAND COPY	Ralph Kenyon	87/4/08
FORTH ad	Ralph Kenyon	81/5/05	HELP COMMAND DELETE	Ralph Kenyon	88/3/09
FORTRAN	Ralph Kenyon	86/4/08	HELP COMMAND ENABLE	Ralph Kenyon	87/1/07
FPL.GO description	PolyLetter	82/4/06	HELP COMMAND INIT	Ralph Kenyon	86/4/07
FRACTION-METRIC-CONV.BS Program Description		87/5/08	HELP COMMAND LIST	Ralph Kenyon	86/4/08
Franklin's Rule	Murphy	88/6/06	HELP COMMAND PACK	Ralph Kenyon	88/3/09
Free Subscriptions End	Ralph Kenyon	87/6/01	HELP COMMAND Printer	Abstract Systems	89/1/09
FREEZING the Screen	Norm Shimmel	80/3/08			
From the Editor's Screen	Bob Bybee	83/5/01			
Front Panel Answer	Ralph Kenyon	86/6/02			
Front Panel Bytes	Ralph Kenyon	88/3/03			
Front Panel Escape	Russ Hobbs	81/4/08			
Front Panel Lockout	John McGav	82/4/02			
Front Panel Question	Constantin Pavloff	86/6/02			
FTP Bug	Ralph Kenyon	83/3/07			
FTP BugNote	Abstract Systems	89/3/09			
FULL HELP	Charles Thompson	80/4/04			
Function MOD in BASIC	Russ Hobbs	81/4/04			
Games: REVERSE.BS	Ralph Kenyon	86/3/05			
Gfid # Bug	Ralph Kenyon	83/3/07			
Gfid Enter Bug	Ralph Kenyon	83/3/07			
Gilb's Laws of Unreliability	Murphy	88/6/06			
Glitch of the Issue	Ralph Kenyon	86/3/02			
Glitch of the Issue - Charles Stein	Charles Steinhauser	85/6/01			
Glitch of the Issue - Frank Stearns	Charles Steinhauser	85/5/01			
Glitch of the Issue -- Bob Bybee	Charles Steinhauser	85/4/01			
Glitch of the Issue -- Ralph Kenyon	Charles Steinhauser	85/3/01			
Global Search and Change	Charles Thompson	88/3/05			
Gnomus	Ralph Kenyon	87/6/01			
Gnomus	Ralph Kenyon	88/5/06			
GOLD6.BS Program Description		87/5/08			
Graphics	Ralph Kenyon	89/4/05			
Greek on the Poly screen	Bob Bybee	81/3/07			
Gumidge's Law	Murphy	88/6/07			
HANGMAN.BS description	Bob Bybee	81/3/05			
Hard disk	PolyMorphic Systems	81/4/02			
Hard Disk Announcement	PolyMorphic Systems	81/1/02			
Hardware at PolyLetter	Ralph Kenyon	86/4/01			
Hardware Errors: Reseat Chips	Ralph Kenyon	86/6/01			
Harp's Corollary to Estridge's Law	Murphy	88/6/07			
Hash Coding	Ken Williams	80/0/02			
Hash Coding	Mark Sutherland	80/5/08			
Hash Coding Program	Ken Williams	80/5/09			
HAYES.AS Program Description		86/4/09			
HD/18 Report	PolyLetter	83/5/03			
Head Cleaning Warning	Don Noe	81/2/02			
Heapsort ad	Ralph Kenyon	81/5/05			

---

### Advertising

Commercial advertising rates are \$50 for a full page, \$25 for a half page, and \$15 for a quarter page. Anything smaller is \$3.00 per column inch. A column is 3-3/4 inches wide by 10 inches tall. A full page is 7-5/8 inches wide. Noncommercial ads by subscribers are free.

---

PolyMorphic 8813 needs home. Make offer. Conway Spitler, P. O. Box 385, Fillmore, CA 93016-0385

---

Entire PolyMorphic System User Manual, System 88 User's Manual with Exec/96 addendum, & System 88 Operation Essentials On IBM disk. Al Levy, 516-293-8358

---

FOR SALE: PolyMorphic System 8813, 64K, three new double-sided drives (Not double density) (also reads & writes single-sided disks); high resolution PolyGrafix video card (640x240) w/inverse video capability (compatible with all standard graphics programs); Programmable Character Generator with software driver; 15" Sanyo monitor; GRI 777 keyboard; cassette & printer mini-cards; Integral Data Systems "BrighterWriter" printer. Enhanced system software. Excellent condition. Best offer. Ken Lowe, 5936 Zina Circle, West Valley City, UT 84120 (801) 969-7736.

---

FOR SALE: Poly 8810 box with power supply and mother board. \$50 plus shipping. Charles A. Thompson, 2909 Rosedale Avenue, Dallas, Texas 75205-1532, (214)-368-8223.

different ways? "SAVE,NAME" saves the program as a text file. "SAVE;NAME" saves the program as a text file also, but inserts "RUN" after the program to make it start automatically.

**DISKS - MODEMS - PROMS - SOFTWARE - SPELL**

**\*\*\* New Low Prices \*\*\***

- 1. MAXELL diskettes: 5-1/4" 10 hard sector -- \$10 per box.
- 2. Used diskettes: 5-1/4" 10 hard sector -- \$0.50 each.
- 3. Hayes Micromodem 100 (300 baud S-100 internal modem) \$20. (If you don't have a modem this is a cheap way to go.)
- 4. HayesSys modem software (for the Micromodem 100) \$10.
- 5. Abstract Systems Exec (Enhancements & bugs corrected) \$30.
- 6. Abstract Systems Proms (Enhancements & bugs corrected) \$35.
- 7. PolyGlott Library Volumes: \$6 each; 5 or more - \$5 each.
- 8. Hayes Smartmodem 1200B (IBM compatible internal) \$40.

Abstract Systems, etc., 191 White Oaks Road, Williamstown, MA 01267, Phone: (413) 458-3597

(Send \$1.00 for a complete catalog--(free with any order).)

(Make check or money order payable to Ralph Kenyon.)

**Bit Bucket**

Did you know that if you can SAVE a BASIC program as a text file in two

**In This Issue**

Editorial . . . . .	1
Letters . . . . .	1
Managing TSR programs (part 2) . . . . .	1
BASIC Pointers . . . . .	3
Serial vs Parallel . . . . .	3
Announcements . . . . .	4
In the Public Domain - PGL-V-33 . . . . .	4
PolyLetter: The first 10 Years . . . . .	4
Advertisements . . . . .	9
The Bit Bucket . . . . .	10

**Coming Soon**

Conclusion of Managing TSR programs, Poly Meta, More: BASIC for Beginners, PC stuff, System Programmers Notes, Help, BugNotes, Public Domain Software, etc. More of PolyLetter: The first 10 years.

**PolyLetter**  
191 White Oaks Road  
Williamstown, MA 01267  
(413) 458-3597

Address Correction Requested

**FIRST CLASS MAIL**



Ralph E. Kenyon, Jr. EXP: 99#9  
Abstract Systems, etc. 184  
191 White Oaks Road  
Williamstown, MA 01267-2256

© Copyright 1989 by Ralph E. Kenyon, Jr.

PolyLetter Editor and Publisher: Ralph Kenyon. Subscriptions: US \$18.00 yr., Canada \$20.00 yr., Overseas \$25.00 yr., payable in US dollars to Ralph Kenyon. Editorial Contributions: Your contributions to this newsletter are always welcome. Articles, suggestions, for articles, or questions you'd like answered are readily accepted. This is your newsletter; please help support it. Non-commercial subscriber ads are free of charge. PolyLetter is not affiliated with PolyMorphic Systems.

Back volumes of PolyLetter (1980 thru 1989) are available at reduced prices payable in US dollars to Ralph Kenyon. 1 - \$15, 2 - \$28, 3 - \$40, 4 - \$50, 5 - \$59, 6 - \$67, 7 - \$75; Canada add \$3 shipping, Overseas add \$10. Individual back issues are also available (\$3.50, \$4.00, \$5.00).



# PolyLetter



PolyLetter 90/5

Page 1

SEP/OCT 1990

## Editorial

All right, you guys and gals out there in PolyLand -- I need help! Take 2 minutes and jot down on a scrap of paper what you would like to see in PolyLetter. Do it NOW! Send it to me at 191 White Oaks Road, Williamstown, MA 01267.

## Letters

Dear Ralph,

October 12, 1990

A few months ago I bragged about being able to fix most Poly breakdowns. The secret of my success is the inventory of parts, boards, etc. which I have to back up the hardware. Now, I am stumped by a problem with one of my three hard disks which Poly is unable to access. I would appreciate any wisdom that you can offer!

The problem occurred suddenly without warning. The HD had been running fine for a long time. Following an edit mode entry, I got a front panel on a basic program. I've checked for loose connections on both boards, tried to interchange another board and HD unit with the sick equipment -- all to no avail.

Do you know someone who might be able to find and fix the problem? I could send just the board and HD unit or the whole thing.

Are there any other HD's which are compatible with an 8813? I would buy one or I would buy an HD similar to mine. I would be glad to advertise for an operational HD, also functioning CPU boards, printer cards, and RAM boards retrofitted for HD operation.

I am resisting conversion to a PC which may be stupid and stubborn; but I haven't got the time or stomach to undergo the trauma. You might say that my twelve years of Poly experience is a miracle because on

an scale of 1 to 10 my computer ability in comparison with you experts is about 1.

My fax number is the same as that listed [below] if you or anyone else wants to reach me 24 hours a day. I would appreciate and compensate you for assistance. -- James Salinger, 2601 Section Road, Cincinnati, OH 45222, Phone: 513/531-3106

[Jim, Bob Bybee has been manufacturing and selling secondary market HD units for the Poly. He may be able to provide repair service on Poly HD units as well. Other hard disks could be compatible with the Poly provided they have an S-100 host adapter card which does not require DMA. We would have to write the software to interface such a hard disk to the Poly, though. Standard Poly boards are used in a system which can use a hard disk. The only Poly systems which need modified cards are either the Twin system or CP/M. Also, anyone with your experience with the Poly is a lot more than a 1. We lost all the 1's years ago. Anybody hanging on this long has gotta be much more than that! -- Ed.]

## Announcements

PM 2.0

September 26, 1990

I've just completed version 2.0 of PM, the PolyMorphic Emulator program for IBM-PC compatibles. PM/2.0 includes two major enhancements which will speed up its operation. The main emulation loop has been improved through the use of some self-modifying code which allows PM to process "emulated interrupts" in parallel with normal execution, instead of checking for interrupts using extra instructions. This more closely emulates what a "real" 8080 does, and gives a substantial speed improvement.

The bottom line is: on CPU-intensive programs, PM now runs about 40% faster.

This assumes no disk activity, and was measured on a 25 MHz 80386 system. (Slower systems, such as a PC or XT, will exhibit less improvement. They have a higher percentage of overhead time used by screen updating.)

The 40% improvement is valid for all programs written in BASIC or assembly language. But wait... there's more! PM/2.0 also contains an emulation of the NORTH STAR FLOATING-POINT BOARD, which is supported by Poly BASIC. This allows the emulator to pass floating-point number calculations back to the IBM-PC for processing, and it's much faster than letting BASIC do the work all by itself. How much faster? One benchmark ran at THREE TIMES its original speed!

"But my programs don't do much floating-point math," you may say. Well, yes they do. Even a simple FOR/NEXT loop in BASIC does floating-point calculations when it adds 1 to the variable after each loop. In BASIC, all math is done to the numerical precision you specify in a DIGITS statement, or 8 digits by default. So, even programs that aren't real "number crunchers" can benefit. If your program does any transcendental functions like SIN, TAN, or EXP, it will really speed up, since BASIC evaluates these functions using a Taylor series, with lots of additions, subtractions, multiplications, and divisions.

The North Star FPB emulation, like the original North Star board, will operate at 6 to 14 digits of precision. BASIC's built-in floating-point routines can operate at up to 26 digits. PM/2.0 has a way to disable the FPB emulation if you need more than 14 digits.

PM/2.0 is available now, and the price is still \$75. If you've purchased a previous version of PM, you can upgrade for only \$25. There's no need to return your original PM disk. Send your order to: Poly Peripherals, 5011 Brougham Court, Stone Mountain GA 30087, or call (404) 498-3556.

PS: Our company recently acquired some 25 MHz 80486 PC-compatibles. I tested PM/2.0 on one of these and found that it runs 3.6 times as fast as an authentic PolyMorphic 8813! Again, that's a CPU-only benchmark. Given that most programs do at least some disk I/O, and PC disks are

faster than Poly disks (especially with a disk cache running), PM/2.0's overall performance on a '486 should be at least FOUR TIMES as fast as a Poly. (For the curious: Norton's SI on this '486 system showed it to be 40.5 times as fast as an original IBM-PC.)

### 10 Year Index (part 2)

HELP COMMAND DELETE	Ralph Kenyon	88/3/09
HELP COMMAND ENABLE	Ralph Kenyon	87/1/07
HELP COMMAND INIT	Ralph Kenyon	86/4/07
HELP COMMAND LIST	Ralph Kenyon	86/4/08
HELP COMMAND PACK	Ralph Kenyon	88/3/09
HELP COMMAND Printer	Abstract Systems	89/1/09
HELP COMMAND RENAME	Ralph Kenyon	88/2/09
HELP COMMAND Sniff	Ralph Kenyon	88/1/09
HELP COMMAND UNDELETE	Abstract Systems	89/6/09
HELP COMMAND ZAP	Ralph Kenyon	88/3/09
HELP files	Ralph Kenyon	86/4/08
HELP flip	Charles Thompson	80/4/04
HELP fold	Charles Thompson	80/4/04
HELP format	Ralph Kenyon	86/5/05
HELP format bs	Ralph Kenyon	86/5/05
HELP FULL	Charles Thompson	80/4/04
HELP IMAGE	Abstract Systems	85/4/07
HELP LOG/NOLOG	Charles Thompson	80/4/04
HELP NEW	Ralph Kenyon	86/3/06
HELP PROGRAM ARISE	Ralph Kenyon	86/5/06
HELP PROGRAM CLEAN	Ralph Kenyon	88/3/09
HELP PROGRAM COMPARE	Abstract Systems	89/4/10
HELP PROGRAM DIRCOPY	Abstract Systems	89/6/09
HELP PROGRAM PUNCH	Ralph Kenyon	86/5/05
HELP PROGRAM Wait	Ralph Kenyon	88/3/09
HELP PROGRAMS	Ralph Kenyon	86/2/06
HELP SYS	Ralph Kenyon	86/2/05
HELP.GO program description	PolyGlot	89/2/06
Helpful Hints in Layman's Language	Charles Thompson	82/1/08
Helpful Hints in Layman's Language	Chuck Thompson	81/6/04
Helpful Hints in Layman's Language	Chuck Thompson	82/3/08
Helpful Hints in Layman's Language	Chuck Thompson	82/4/04
Helpful Hints in Layman's Language	Chuck Thompson	83/4/08
Helpful Hints in Layman's Language	Chuck Thompson	85/6/02
Helpful Hints in Laymen's Language	Charles Thompson	86/1/06
HEX INPUT/OUTPUT in BASIC	PolyLetter	83/1/07
HEX to decimal conversion	Mike Stanford	80/5/04
Hidden Messages	Al Levy	86/3/07
Hidden Messages	Ralph Kenyon	86/3/07
Hinds' Law of Computer Programming	Murphy	88/6/07
Historical Sketch of Poly	Ralph Kenyon	86/3/01
History of 8080 Emulation	Bob Bybee	89/5/06
Hoare's Law of Large Programs	Murphy	88/6/07
HORSE-RACE.BS Program Description		86/5/09
Hot-Key	Ralph Kenyon	87/6/01
HOURS/DEMO.BS Program Description		87/5/08
How about "REAL" CP/M	PolyLetter	82/6/04
How BASIC Stores Variables	Bob Bybee	80/5/05
How Computers can make you rich	Charles Gross	81/5/03
How FORMAT.GO Works	Ralph Kenyon	83/4/06



How it Works -- The Video Terminal	Bob Bybee	84/5/01	Key Bounce Problem	Constantin Pavloff	86/6/02
How it Works -- Disk Controllers	Bob Bybee	85/2/02	Key Bounce Solution	Ralph Kenyon	86/6/02
How it Works -- Disk Controllers	Bob Bybee	85/2/02	Key Cap Puller	Ralph Kenyon	87/1/07
How Many Polys are there	Ralph Kenyon	83/3/08	Key Cap Removal	Ralph Kenyon	87/1/07
How Programs Run on the Poly, etc	Bob Bybee	85/5/04	Keyboard Cover	Ralph Kenyon	89/1/05
How to Compile Small C Programs	Ron Cain	88/4/05	Keyboard Key Cleaning	Ralph Kenyon	86/6/02
How to erase yourself	Bob Felts	81/6/09	Keyboards and WordMaster II	Ralph Kenyon	86/3/05
How to Fix A Dead Bird	Charles Steinhauser	87/4/06	Keyboard speedup	Frank Stearns	81/4/04
How to get your PC to PAGE	Ralph Kenyon	88/1/09	Keys Sticking	Ralph Kenyon	87/1/07
How to SAVE 'SAVE'	Mark Sutherland	80/3/02	Kind Words	Michael Aquino	85/1/02
How to ship disks	PolyLetter	85/1/04	KISS.RL	ISS	82/1/04
How to UNSAVEP (part 1)	Ralph Kenyon	88/2/06	Languages	Ralph Kenyon	88/2/01
How to UNSAVEP (part 2)	Ralph Kenyon	88/3/07	Languages	Ralph Kenyon	88/3/01
How to UNSAVEP (part 3)	Ralph Kenyon	88/4/09	LEMONADE-STAND.BS Program Description		86/5/09
How to UNSAVEP (part 4)	Ralph Kenyon	88/5/02	LEN input: BASIC: BugNote 008.0	Ralph Kenyon	86/6/07
How to use Mailist	Al Levy	85/4/05	Letter	Al Levy	86/2/02
How to use Mailist	Al Levy	85/5/02	Letter	Al Levy	86/3/01
How to use Mailist -- Preface	Al Levy	85/6/03	Letter	Al Levy	86/3/02
How To Use Setup	Charles Steinhauser	86/1/03	Letter	Allen Daubendiek	88/5/01
Hsrt.OV information	Ralph Kenyon	87/5/03	Letter	Allen Daubendiek	89/3/01
IBM	Al Levy	86/2/04	Letter	Arnell Baker	89/6/01
IBM	Al Levy	87/1/06	Letter	Barkley Gilpin	87/2/01
IBM BASIC	Ralph Kenyon	88/2/09	Letter	Bob Bybee	85/6/02
IBM BASIC	Ralph Kenyon	88/3/07	Letter	Bob Bybee	87/2/05
IBM made to LIST like Poly	Ralph Kenyon	86/6/09	Letter	Bob Bybee	87/6/03
IBM made to TYPE like Poly	Ralph Kenyon	86/4/03	Letter	Bob Bybee	88/5/01
IBM PC	PolyLetter	82/2/03	Letter	Bob Johnson	87/2/01
IBM Program Exchequer	Ralph Kenyon	89/1/07	Letter	Bob Johnson	87/3/06
IBM Program TICKLER.BAS	Ralph Kenyon	88/5/03	Letter	Bob Jones	87/1/04
IBM Program TICKLER.BAS listing	Ralph Kenyon	88/5/05	Letter	Charles Mach	86/5/03
Ideas from Russ Hobbs	PolyLetter	84/6/01	Letter	Charles Mach	86/6/01
IDS Printer Interface	Ralph Kenyon	81/2/04	Letter	Charles Mach	86/6/03
IMAG - remove system disk	Chuck Thompson	81/6/04	Letter	Chris Bagley	86/5/03
IMAGE non system disks	Dan DeForest	80/3/09	Letter	Chris Bagley	87/1/02
INSA11	Bob Bybee	81/5/03	Letter	Chuck Gross	86/5/02
Increasing BASIC precision	John McGav	81/3/02	Letter	Chuck Gross	87/5/03
Index - 1981	PolyLetter	82/1/04	Letter	Chuck Thompson	87/2/04
Index of BUG HELP	Abstract Systems	89/2/09	Letter	Chuck Thompson	87/5/01
Index to PolyLetter 1980	PolyLetter	81/1/03	Letter	Constantin Pavloff	86/6/02
Index to PolyLetter 1982	PolyLetter	83/1/02	Letter	Dana Gould	87/4/04
INDUSTRIAL-SIMULATION.BS Program Description		86/5/09	Letter	Dick Brown	87/3/01
INIT system disk	Chuck Thompson	81/6/04	Letter	Doug Henry	86/4/02
INPUT Bug in BASIC	Ralph Kenyon	83/1/03	Letter	Doug Schirripa	85/2/01
INPUT.BS program description	Bob Bybee	81/4/05	Letter	Dr. Briarty	87/1/02
Input: Assembly Language - WHO	Ralph Kenyon	86/4/06	Letter	Earl Gilbreath	87/2/03
Interview with Don Moe	Bob Bybee	81/2/01	Letter	Earl Gilbreath	87/5/02
Introduction to C	Bob Bybee	88/3/01	Letter	Earl Gilbreath	86/3/02
Inventory	Bob Johnson	87/2/01	Letter	Earl Gilbreath	89/1/01
Inventory Available	Percy Roy	86/5/01	Letter	Earl Gilbreath	89/3/01
INVENTORY Program		87/2/07	Letter	F. Marc de Piolence	88/1/01
Inventory System Wanted	Travis Miller	86/3/05	Letter	Frank Stearns	87/4/02
Is There Any Hope For Poly Users	Al Levy	86/2/03	Letter	Frank Stearns	87/5/03
Is Your Poly Covered?	PolyLetter	82/5/04	Letter	Gary Sterling	87/3/04
JAN 81 Disk of Month Listing	PolyLetter	84/1/06	Letter	Glenn McComb	87/2/05
JUL 81 Disk of Month Listing	PolyLetter	84/1/05	Letter	Glenn McComb	87/3/06
JUL 81 DOM discription	Bob Bybee	81/4/05	Letter	Jack Hills	86/5/02
JUL 82 Disk of Month listing	PolyLetter	82/4/06	Letter	James Purvis	86/5/01
JUL 82 Disk of Month Listing	PolyLetter	84/1/05	Letter	James Purvis	86/6/03
JUL 82 Disk of Month Listing	PolyLetter	84/1/05	Letter	Jim Ryan	86/5/03
JUL 83 Disk of Month listing	PolyLetter	83/4/04	Letter	Jim Salinnger	86/3/02

Letter	Jim Trahan	86/4/01	Machine Language Calls from BASIC	80/3/03
Letter	Jim Trahan	87/3/01	Machine Language Programming	Bob Bybee 80/1/02
Letter	John McHally	86/4/01	Machine Language Sorts Review	Ken Williams 80/3/09
Letter	John Sloan	87/2/04	Macintosh	Michael Aquino 87/5/06
Letter	Jose Lipana	87/2/01	Macro's Anyone	Ralph Kenyon 85/6/04
Letter	Joseph Toman	86/5/02	MACRO-85	PolyMorphic Systems 81/5/02
Letter	Ken Love	89/4/01	Macros, Anyone?	PolyLetter 82/6/09
Letter	Kenneth Lowe	88/5/01	Magnetic Disk Holder	Mark Sutherland 80/2/05
Letter	Len Thomsen	87/3/03	MAILER.AS Program Description	86/4/09
Letter	Len Thomsen	87/5/07	Mailist, Using - Part 4	Al Levy 86/1/02
Letter	Mark de Piolenc	87/3/05	Maintenance Hints BASIC Software	Ralph Kenyon 86/5/08
Letter	Michael Aquino	85/2/04	MAJOR-MINOR-FINDER.BS Program Description	86/5/09
Letter	Michael Aquino	86/6/04	MAKEREL	Bob Bybee 87/6/03
Letter	Michael Aquino	87/5/06	MakeRel	Ralph Kenyon 87/6/04
Letter	Michael Schwartz	87/5/03	Map of Adventure Maze	Ralph Kenyon 86/2/07
Letter	Michell Lippman	87/2/01	MAR 81 Disk of Month Listing	PolyLetter 84/1/05
Letter	Norm Shimmel	87/2/02	MAR 82 Disk of Month Listing	PolyLetter 84/1/05
Letter	Percy Roy	86/5/01	MAR 83 Disk of Month listing	PolyLetter 83/2/06
Letter	Percy Roy	88/5/01	MAR 83 Disk of Month Listing	PolyLetter 84/1/05
Letter	Ralph Kenyon	85/4/02	March 82 Disk of Month	PolyLetter 82/2/05
Letter	Richard Rodman	87/3/03	MASTERMIND.BS description	PolyLetter 82/4/06
Letter	Richard Wagner	89/4/01	MAT PRINT Bug in BASIC C03	Ralph Kenyon 82/1/06
Letter	Robert Johnson	86/5/03	MAT WRITE Bug in BASIC C03	Ralph Kenyon 82/1/06
Letter	Robert Schwartz	85/2/06	MAT WRITE File Channel bug	Ralph Kenyon 82/1/06
Letter	Robert Schwartz	87/2/01	MATCH-POINT	Ralph Kenyon 86/2/05
Letter	Robert Schwartz	87/2/02	MAY 81 Disk of Month Listing	PolyLetter 84/1/05
Letter	Robert Schwartz	87/4/01	MAY-81-DEM description	PolyLetter 81/3/05
Letter	Robert Stricklin	89/3/01	Maze Map for Adventure	Ralph Kenyon 86/2/07
Letter	Ron Moffatt	87/3/03	MDIR.BS Program Description	87/5/08
Letter	Ron Moffatt	87/4/04	Medical Programs Anyone?	John Sparti, M.D. 80/1/03
Letter	Ron Moffatt	87/6/03	Megabytes and Megabytes	Bob Bybee 80/5/01
Letter	Russ Nobbs	85/2/01	Memory board bug - 16K	Mark Sutherland 80/4/02
Letter	Russ Nobbs	86/4/01	Memory Management Techniques	Bob Bybee 88/4/06
Letter	Wayne Devey	87/4/04	Memory Resident programs	Bob Bybee 88/4/07
Letter Formatting in BASIC	Bill Sullivan	81/5/07	Memory Resident Programs	Ralph Kenyon 88/5/05
Letter to Poly	Russ Nobbs	81/6/01	Memory-Resident	Ralph Kenyon 87/6/01
Lifeboat Abandons CP/M	Mark Sutherland	80/2/03	MENTEST.AS - A Memory Test Program	Ralph Kenyon 86/4/06
LINE-FEED Generation Bypass	Ralph Kenyon	86/5/06	Merging Mailing Lists with Format	Ralph Kenyon 82/2/06
LINEAR.BS Program Description		87/5/08	Meskimen's Law	Murphy 88/6/07
LIST on a Clone	Ralph Kenyon	86/6/09	Microline 80 printer	PolyLetter 82/4/05
Listing - APR-80 Disk of Month		80/2/06	Micropolis Users Group	PolyLetter 82/1/03
Listing - AUG-80 Disk of Month		80/4/08	MKDIR.GO description	PolyLetter 83/4/04
Listing of PGL-V-26	Ralph Kenyon	88/1/07	MM100.AS Program Description	86/4/09
Listing of PGL-V-27	Ralph Kenyon	88/2/06	MOD function in BASIC	Russ Nobbs 81/4/04
Listing of PGL-V-28	Ralph Kenyon	88/3/06	Modem software	Ralph Kenyon 89/3/08
Little Known Error Messages	Mark Sutherland	80/3/07	Modem-kit ad	Bob Bybee 81/5/05
Little-ADA ad	Ralph Kenyon	81/5/05	MODEM-T.GO program description	Ralph Kenyon 89/3/08
Little-Ada description	Ralph Kenyon	88/2/06	Modems	Ralph Kenyon 89/3/05
Little-Ada syntax diagrams	Ralph Kenyon	88/2/07	Modems and Communications	PolyLetter 83/5/04
Load Addresses and Start Addresses	PolyLetter	83/5/03	Modems and Communications Part 2	Bob Bybee 83/6/05
Local Area Net for Poly	Bob Bybee	88/1/01	Modems, More on	Bob Bybee 89/4/07
LOG/NOLOG HELP	Charles Thompson	80/4/04	Modernizing your Poly	Bob Bybee 85/2/04
Logic Families	Bob Bybee	89/1/04	Modernizing your Poly	Bob Bybee 85/4/01
Lomas 80386 S-100 boards	Al Levy	87/1/05	MOON-LANDER.BS program description	Bob Bybee 81/6/05
Look and Feel Lavsuits	Bob Todd	88/6/05	More About Adventure	Ralph Kenyon 85/5/08
Loop.GO program description	PolyGlot	89/2/06	More C	Ralph Kenyon 88/4/01
Lost Poly Users	Ralph Kenyon	86/4/09	More on cleaning MS drives	PolyMorphic Systems 81/4/03
Lost Poly Users	Ralph Kenyon	86/5/09	More on Confidence and MENTOP	PolyLetter 82/3/04
Low Voltage problem	PolyLetter	84/1/04	More on Freezing the Screen	Russ Nobbs 81/6/03
LREGRS.BS Program Description		87/5/08	More on Macros	PolyLetter 83/6/04



More on Modems	Bob Bybee	89/4/07	Operate Your Poly Remotely	Bob Bybee	80/5/06
More on ROM versions	PolyLetter	82/2/02	Operating System Revisions	PolyLetter	82/1/03
More on Using Format from Exec	Russ Nobbs	82/5/04	Operating System Revisions	PolyLetter	82/3/03
MORSE-CODE.BS Program Description		86/5/09	OREGON.BS Program Description		87/4/07
MOVE	Bob Bybee	81/1/05	OS/VU	PolyLetter	81/4/06
Move.GO program description	PolyGlot	89/2/06	Osborn's Law	Murphy	88/6/07
Moving Subdirectories	Ralph Kenyon	88/6/02	OTHELLO.BS Program Description		87/4/07
MS	Mark Sutherland	80/1/04	Other Guys	Ralph Kenyon	86/4/03
MS Controler Ports BugNote 007.0	Ralph Kenyon	86/5/08	Other Guys	Ralph Kenyon	86/6/09
MS Drives - Cleaning	PolyMorphic Systems	81/4/03	OUT 0, OUT 0, DAMNED SPOT	Norm Shimmel	88/1/08
MS Error messages	Russ Nobbs	81/4/06	OUT BUG - BugNote 24	Ralph Kenyon	88/1/09
MS Unit Types	PolyLetter	82/4/08	Output: Assembly Language - WH1	Ralph Kenyon	86/4/06
MSDOS	Al Levy	86/2/02	PACK Bug	Ralph Kenyon	83/1/03
Msg (System Programmers Guide)	Ralph Kenyon	88/3/09	PACK bug - Exec/90	PolyLetter	81/6/03
Murphy'S Computer Law	Ralph Kenyon	88/6/06	Pack Error: NO Abort: BugNote 009.0	Ralph Kenyon	86/6/07
Murphy's Fourth Law	Murphy	88/6/07	PACK HELP	Ralph Kenyon	88/3/09
Murphy's Law of Thermodynamics	Murphy	88/6/07	PACK in command files	Al Levy	82/2/02
Murphy's Laws of Computers	PolyLetter	81/1/02	Pack Syntax Failure BugNote 005.0	Ralph Kenyon	86/5/08
My Favorite DOS programs	Al Levy	89/2/01	PAGE gives TITLE	Ralph Kenyon	80/3/09
Name That Disk	PolyLetter	82/5/05	PANEL.GO program description	PolyGlot	89/2/06
Naming Programming Languages	Ralph Kenyon	86/5/09	Parallel 8251 Schematic	John Warkentin	87/6/07
Networking	Bob Bybee	86/4/03	Parallel 8251 UART	John Warkentin	87/6/06
New Operating System	PolyLetter	81/4/06	Parity	Ralph Kenyon	87/1/03
New Products from Poly	PolyMorphic Systems	81/6/01	PASCAL	Ralph Kenyon	89/3/01
New ROM for 88-MS	PolyLetter	83/6/03	Pascal Pastures	Bob Bybee	84/2/02
New System	Ralph Kenyon	86/4/04	Pascal Pastures	PolyLetter	84/4/02
New Wordmaster	Mark Sutherland	80/4/01	Pascal Wanted	Percy Roy	86/5/01
News	PolyLetter	84/2/01	PAYROLL.BS Program Description		87/5/08
News	PolyLetter	84/3/01	PCALC	Bob Bybee	86/4/03
News	PolyLetter	84/4/01	PcId.OV	Bob Bybee	88/4/09
News	PolyLetter	84/5/01	PCOPY.GO	PolyLetter	83/5/04
News	PolyLetter	84/6/01	PEEK(0)	PolyLetter	81/1/01
News from PolyMorphic	Bob Bybee	82/1/01	PEEK(1)	PolyLetter	81/1/01
News from PolyMorphic	Bob Bybee	83/1/01	PEEK(2)	PolyLetter	81/1/01
News from PolyMorphic	Bob Bybee	83/2/01	PEEK(3) & PEEK(4)	PolyLetter	81/1/01
News from PolyMorphic	Bob Bybee	83/3/01	PEEK(5)	PolyLetter	81/1/01
News from PolyMorphic	PolyLetter	83/6/02	PEEK(9)	PolyLetter	81/1/01
News from PolyMorphic System	PolyLetter	85/3/01	PEEK-DUMP.BS Description	PolyLetter	81/2/08
News from PolyMorphic Systems	PolyLetter	82/3/01	PEEKER.BS Program Description		87/5/08
News from PolyMorphic Systems	PolyLetter	82/4/01	Peer's Law	Murphy	88/6/07
News from PolyMorphic Systems	PolyLetter	82/5/01	People	Bob Bybee	88/6/08
News from PolyMorphic Systems	PolyLetter	82/6/01	Perfect Calc	Bill Davis	83/4/05
Ninety-Ninety Rule of Project Schedules	Murphy	88/6/07	PERMUTATIONS.BS program	Bob Felts	81/4/06
Nixon's Theorem	Murphy	88/6/07	PGL-V-01 listing	PolyGlot	89/2/07
No 8 bits from the Serial Port	PolyLetter	84/2/05	PGL-V-02 listing	PolyGlot	89/2/07
No CP/M for PolyMorphic	Mark Sutherland	80/1/01	PGL-V-03 listing	PolyGlot	89/2/07
Nolan's Placebo	Murphy	88/6/07	PGL-V-04 listing	PolyGlot	89/2/07
Non-system IMAGE	Dan DeForest	80/3/09	PGL-V-05 listing	PolyGlot	89/2/07
Notes from Poly	PolyLetter	81/5/01	PGL-V-06 listing	PolyGlot	89/2/07
NOV 81 DOM description	Bob Bybee	81/6/05	PGL-V-07 listing	PolyGlot	89/2/07
NUMBER-CONVERSION.BS Program Description		86/5/09	PGL-V-08 listing	PolyGlot	89/2/08
O'Toole's Commentary on Murphy's Law	Murphy	88/6/07	PGL-V-09 listing	PolyGlot	89/2/08
Odds and Ends from Atlanta	Bob Bybee	86/4/03	PGL-V-10 Listing		87/1/08
Okidata 192 Printer	Earl Gilbreadth	87/2/07	PGL-V-10 listing	PolyGlot	89/2/08
ON ERROR	PolyLetter	82/3/05	PGL-V-11 Listing		87/1/08
ON ERROR Bug	PolyLetter	83/2/07	PGL-V-11 listing	PolyGlot	89/2/08
ON ESCAPE	PolyLetter	82/3/05	PGL-V-12 Listing		87/1/08
On the FORTH day of ...	Ralph Kenyon	81/5/07	PGL-V-12 listing	PolyGlot	89/2/08
One Day in the Life of [A:S] Poly	Ralph Kenyon	89/6/01	PGL-V-13 Listing		87/1/08
OOPS! SYSTEM DISK OUT	Bob Bybee	80/4/06	PGL-V-13 listing	PolyGlot	89/2/08

PGL-V-14 Listing		07/1/08	PolyGlott Library Volume 01 Listing		08/2/07
PGL-V-14 listing	PolyGlott	09/2/08	PolyGlott Library Volume 02 Listing		08/2/07
PGL-V-15 Listing		07/1/08	PolyGlott Library Volume 03 Listing		08/2/07
PGL-V-15 listing	PolyGlott	09/2/08	PolyGlott Library Volume 04 Listing		08/2/07
PGL-V-16 Listing		07/1/08	PolyGlott Library Volume 05 Listing		08/2/07
PGL-V-16 listing	PolyGlott	09/2/08	PolyGlott Library Volume 06 Listing		08/2/07
PGL-V-17 listing		07/2/07	PolyGlott Library Volume 07 Listing		08/4/08
PGL-V-17 listing	PolyGlott	09/2/08	PolyGlott Library Volume 08 Listing		08/4/09
PGL-V-18 listing		07/2/08	PolyGlott Users Group	Ralph Kenyon	03/4/02
PGL-V-18 listing	PolyGlott	09/2/08	PolyLetter Back Issue Prices		07/3/06
PGL-V-19 listing		07/2/08	PolyLetter Hardware	Ralph Kenyon	08/4/01
PGL-V-19 listing	PolyGlott	09/2/08	PolyLetter has Time Sharing	Mark Sutherland	00/3/01
PGL-V-20 listing		07/2/08	PolyLetter Printing	Ralph Kenyon	08/4/01
PGL-V-20 listing	PolyGlott	09/2/08	PolyLetter Subscriber Address List	Charles Steinhauser	08/1/04
PGL-V-21 listing		07/2/08	PolyLetter Subscriber List	PolyLetter	03/1/08
PGL-V-21 listing	PolyGlott	09/2/08	PolyLetter under New Management	Frank Stearns	03/6/01
PGL-V-22 listing		07/2/08	PolyLetter Visits PolyMorphic	Mark Sutherland	00/3/04
PGL-V-22 listing	PolyGlott	09/2/08	PolyLost	Ralph Kenyon	08/4/09
PGL-V-23 listing		07/4/07	PolyLost	Ralph Kenyon	08/5/09
PGL-V-23 listing	PolyGlott	09/2/08	PolyMorphic Systems	Bob Bybee	08/3/02
PGL-V-24 Listing		07/5/08	PolyMorphic Systems	Ralph Kenyon	08/2/05
PGL-V-24 listing	PolyGlott	09/2/08	PolyMorphic Systems	Ralph Kenyon	08/3/01
PGL-V-25 Listing		07/6/08	PolyMorphic Systems Moves	PolyLetter	01/6/01
PGL-V-25 listing	PolyGlott	09/2/09	PolyMorphic Users Group		00/1/02
PGL-V-26 listing	PolyGlott	09/2/09	PolyMorphic Visit's Georgia	Mark Sutherland	00/1/05
PGL-V-26 Listing	Ralph Kenyon	08/1/07	PolyTalk - User Requests		00/2/05
PGL-V-27 listing	PolyGlott	09/2/09	Port "0" Serial Card Upgrade	Ralph Kenyon	05/4/04
PGL-V-27 listing	Ralph Kenyon	08/2/06	Portability	PolyLetter	02/6/07
PGL-V-28 listing	PolyGlott	09/2/09	POUNCE.BS Program Description		07/4/07
PGL-V-28 listing	Ralph Kenyon	08/3/06	pram.GO program description	PolyGlott	09/2/06
PGL-V-29 description - FORTH	PolyGlott	09/1/09	PramKey	Ralph Kenyon	07/6/01
PGL-V-29 listing	PolyGlott	09/2/09	Pretty Little PolyNomial	Harold C. Kinne	00/4/03
PGL-V-30 Discription	PolyGlott	09/2/05	Prices of PolyLetter Back Issue		07/3/06
PGL-V-31 description (Spell.GO)	Abstract Systems	09/4/10	Primitive I/O in IBM BIOS	Ralph Kenyon	09/6/03
Philosophy of MORE (memory)	Ralph Kenyon	08/4/01	PRINT AT	Lon Amick	00/3/07
PINE	Bob Bybee	08/1/01	PRINT Bug in BASIC C03 using MAT	Ralph Kenyon	02/1/06
Plotter: Trouble Using	John McHally	08/4/01	Print-to-a-file program ad	Ralph Kenyon	01/4/05
Plug for Printer Interface Test	Russ Nobbs	06/6/07	Printer command HELP	Abstract Systems	09/1/09
PMU.GO program description	Bob Bybee	09/5/04	Printer Driver /42	Patty Ryan (PM)	01/2/03
POKE 0,X	PolyLetter	01/1/01	Printer driver bypass	Chuck Thompson	01/3/05
POKE 6,X & POKE 7,X	PolyLetter	01/1/02	Printer Interface Confidence Test	Russ Nobbs	08/6/07
Poly -- Four Years Later	Frank Stearns	02/5/08	Printer POKEs	Bob Bybee	00/5/03
Poly Closing!	Ralph Kenyon	08/6/01	Printer spooler ad	Bob Bybee	01/5/05
Poly Dictionary	ISS	01/5/03	Printer/30	Ralph Kenyon	07/1/03
Poly Disk Drive Upgrade	Charles Steinhauser	04/6/03	Printers Cheap	Bob Bybee	08/4/03
Poly Emulator	Bob Bybee	09/5/01	Printing PolyLetter	Ralph Kenyon	08/4/01
Poly Emulator evaluation	Ralph Kenyon	09/5/08	Prnt.OV description	Chuck Thompson	02/3/08
Poly Graphics	Ken Love	05/1/01	Processing BASIC Errors	John Warkentin	01/2/03
Poly Merger called off	PolyLetter	01/3/01	Program Abort.GO description	PolyGlott	09/2/06
Poly Problems	Bob Bybee	07/1/01	Program ACCURATE-DIVIDER.BS Description		08/5/09
Poly Users List	PolyLetter	09/1/02	Program ARTIL.BS description	Bob Bybee	01/6/05
Poly Users on the Source	PolyLetter	01/2/02	Program BACKGAMMON.BS description	Bob Bybee	01/6/05
Poly Users on the Source	PolyLetter	02/5/03	Program BACKUP.GO description	Chuck Thompson	01/6/04
Poly Users on the Source	PolyLetter	03/2/02	Program BASES.GO description	PolyLetter	03/2/06
Poly Utility in 1984	PolyLetter	04/1/01	Program BASIC-USER-FUNCTION-TABS.BS Description		08/5/09
Poly's Best Kept Secret	Ralph Kenyon	07/6/01	Program BASIC-USER-FUNCTIONS.BS Description		08/5/09
Poly, A Conversation With	Bob Bybee	08/3/02	Program BATTLESHIP.BS description	Bob Bybee	01/3/05
Poly, Historical Sketch of	Ralph Kenyon	08/3/01	Program BIORHYTHM-1.BS Description		08/5/09
Poly-88 Users Group	PolyLetter	01/5/02	Program BIORHYTHM-2.BS Description		08/5/09
Poly-88 Users Group	Roger Lewis	01/2/05	Program BIORHYTHM.BS description	Bob Bybee	01/3/05



Program BOWLING.BS description	PolyLetter	82/4/06	Program Select.GO ad	Ralph Kenyon	81/4/05
Program CHANGE.GO description	PolyLetter	82/4/06	Program Select.GO description	PolyGlot	89/2/06
Program CHESS.GO description	Bob Bybee	81/3/05	Program SETNEW.GO description	Chuck Thompson	81/6/04
PROGRAM CLEAN HELP	Ralph Kenyon	88/3/09	Program Sex-Appeal.BS description	Bob Bybee	81/6/05
Program CLEARNEW.GO description	Chuck Thompson	81/6/04	Program SLOT.BS description	Bob Bybee	81/6/05
Program CMDF.AS Description		86/4/09	Program SM.GO description	Ralph Kenyon	89/3/08
Program COMP-DISK.GO description	Chuck Thompson	81/6/05	Program SNIFFALL.GO description	PolyLetter	83/2/06
PROGRAM COMPARE HELP	Abstract Systems	89/4/10	Program SORT-BUBBLE.BS Description		86/5/09
Program COUNT.GO description	Bob Bybee	81/4/05	Program SORT-HEAP.BS Description		86/5/09
Program DATA-ENTRY.BS description	PolyLetter	83/2/06	Program Space.AS Description		86/4/09
Program dial.GO description	Ralph Kenyon	89/3/09	Program SPACE.AS Description		86/4/09
PROGRAM DIRCOPY HELP	Abstract Systems	89/6/09	Program STAR-LANES.BS Description		86/5/09
Program DIS80.GO description	PolyLetter	83/4/04	Program Submit.GO	Ralph Kenyon	88/1/02
Program DisAsmb.BS ad	Ralph Kenyon	81/4/05	Program SUN-RISE/SUN-SET.BS Description		86/5/09
Program DX.GO description	PolyLetter	82/4/06	Program SYSTEM-SORT-PRINT.BS Description		86/5/09
Program edit description	Ralph Kenyon	88/5/07	Program TABBER.BS Description	PolyLetter	81/2/08
Program ERROR.GO description	PolyLetter	83/4/04	Program TAPE-LOAD.AS Description		86/4/09
Program FIND.GO description	PolyLetter	82/4/06	Program TELETYPE-CODE.BS Description		86/5/09
Program FLIES.BS description	Bob Bybee	81/4/05	Program TERMINAL-FC.AS Description		86/4/09
Program FNTIMER.BS Description	PolyLetter	81/2/08	Program TEXT-TRAN.BS description	PolyLetter	82/4/06
Program FPL.GO description	PolyLetter	82/4/06	Program TICKLER.BS listing	Ralph Kenyon	88/5/03
Program Gnomus description	Ralph Kenyon	88/5/06	PROGRAM to Fold Strings in BASIC	Mark Sutherland	80/3/03
Program HANGMAN.BS description	Bob Bybee	81/3/05	Program Top of Memory Is	John Warkentin	80/2/05
Program HAYES.AS Description		86/4/09	Program Tran.OV Description	PolyLetter	81/2/08
Program HELP.GO description	PolyGlot	89/2/06	Program Tweak.AS Description		86/4/09
Program HORSE-RACE.BS Description		86/5/09	Program VOL.GO description	PolyGlot	89/2/07
Program INDUSTRIAL-SIMULATION.BS Description		86/5/09	Program VSelect.GO ad	Ralph Kenyon	81/4/05
Program INPUT.BS description	Bob Bybee	81/4/05	Program VSelect.GO description	PolyGlot	89/2/07
Program LEMONADE-STAND.BS Description		86/5/09	PROGRAM Wait HELP	Ralph Kenyon	88/3/09
Program Loop.GO description	PolyGlot	89/2/06	Program Wait.AS Description		86/4/09
Program MAILER.AS Description		86/4/09	PROGRAM-LOAD.AS Program Description		86/4/09
Program MAJOR-MINOR-FINDER.BS Description		86/5/09	Programmers Guide	Ralph Kenyon	86/4/06
Program MASTERMIND.BS description	PolyLetter	82/4/06	Programming Languages: Naming	Ralph Kenyon	86/5/09
Program MKDIR.GO description	PolyLetter	83/4/04	PROGRAMS: HELP	Ralph Kenyon	86/2/06
Program MM100.AS Description		86/4/09	PROMS Dio Read Error BugNote 004.0	Ralph Kenyon	86/4/08
Program MODEM-T.GO description	Ralph Kenyon	89/3/08	Protect Part of Your Screen	Mark Sutherland	80/1/01
Program MOON-LANDER.BS description	Bob Bybee	81/6/05	Public Domain	Ralph Kenyon	86/4/08
Program MORSE-CODE.BS Description		86/5/09	Public Domain	Ralph Kenyon	86/5/09
Program Move.GO description	PolyGlot	89/2/06	Put ESC into a Text File	Al Levy	86/1/02
Program NUMBER-CONVERSION.BS Description		86/5/09	Putting ESC Into a Text File	Ralph Kenyon	86/3/03
Program PANEL.GO description	PolyGlot	89/2/06	Qsrt.OV information	Ralph Kenyon	87/5/03
Program PEEK-DUMP.BS Description	PolyLetter	81/2/08	Quick Reference Card		87/2/10
Program PMU.GO description	Bob Bybee	89/5/04	Quicksort ad	Ralph Kenyon	81/5/05
Program pram.GO description	PolyGlot	89/2/06	Radio Shack DWP 230 Printer	Ralph Kenyon	87/5/03
Program Print-to-a-file ad	Ralph Kenyon	81/4/05	RDE.AS Program Description		86/4/09
Program PROGRAM-LOAD.AS Description		86/4/09	READ Bug in BASIC	Ralph Kenyon	83/1/03
Program RDE.AS Description		86/4/09	READ.GO program description	Bob Bybee	81/4/05
Program READ.GO description	Bob Bybee	81/4/05	Reader's Responses	George Montillon	86/6/09
Program Rebuild.GO ad	Ralph Kenyon	81/4/05	Reader's Responses	James Goodman	86/6/09
PROGRAM Redirect Screen to Printer	Mark Sutherland	80/3/02	Readers Requests		87/6/09
PROGRAM RENERR.BS	Norm Shimmel	80/4/07	Readers' Responses	Ralph Kenyon	87/3/09
Program Replace.GO ad	Ralph Kenyon	81/4/05	REBEL.BS Program Description		87/4/08
Program ReStart.GO description	PolyGlot	89/2/06	Rebuild.GO program ad	Ralph Kenyon	81/4/05
Program Restore.GO description	PolyGlot	89/2/06	Recovering from Edit disaster	Chuck Thompson	81/3/04
Program SCAN.GO description	PolyLetter	83/2/06	Redirect Screen to Printer	Mark Sutherland	80/3/02
Program SCOPY.GO description	Chuck Thompson	81/6/05	REENTER not allowed now!	Ralph Kenyon	88/6/05
Program SCREEN-PROTECTED-DEMO.BS Description		86/5/09	REENTRY made easy	Ralph Kenyon	81/6/09
Program SDIR.GO description	PolyLetter	82/4/06	Relocatable Files	Ralph Kenyon	89/6/04
Program Search.GO description	PolyGlot	89/2/06	Remote system ad	Bob Bybee	81/5/05
Program SEAWAR.BS Description		86/5/09	REMS/LINES/OUT.BS Program Description		87/5/08

RENAME Bug	Ralph Kenyon	83/1/03	SLOT.BS program description	Bob Bybee	81/6/05
RENAME HELP	Ralph Kenyon	88/2/09	SM.GO program description	Ralph Kenyon	89/3/08
RENAME with "*" bug BugNote 002.0	Ralph Kenyon	86/3/06	Small Print with a Diablo	Michael Aquino	86/6/04
RENERR.BS PROGRAM	Norm Shimmel	80/4/07	Small-C		87/1/08
Replace.GO program ad	Ralph Kenyon	81/4/05	Small-C Documentation	Ron Cain	88/4/05
Report on the HD/18	PolyLetter	83/5/03	Smith Corono TP-1 Printer	PolyLetter	82/4/05
Reports on Exec/96	PolyLetter	83/3/01	Smoke Test	Ralph Kenyon	87/3/07
Resale Value of a Poly in 1984	Russ Hobbs	86/3/04	Snafu Equations	Murphy	88/6/07
Reseat Chips	Ralph Kenyon	86/6/01	SMIFFALL.GO description	PolyLetter	83/2/06
ReStart.GO program description	PolyGlot	89/2/06	Software Boots	PolyLetter	84/5/04
Restore.GO program description	PolyGlot	89/2/06	Software Packages Available	PolyLetter	82/1/02
REVERSE.BS - A BASIC Game	Ralph Kenyon	86/3/05	Software Shop	Bob Bybee	81/5/05
Review of Anchor Signalman Modeb	PolyLetter	85/2/03	Software Shop	PolyLetter	81/2/04
Review of DisAsmb.BS	Mark Sutherland	80/2/03	Software Source: Buying Poly's	Bob Bybee	86/3/02
Review of the Toshiba P1351 Printer	John J. Warkentin	84/4/03	SORT TIP	Bob Bybee	80/5/13
Review of two Sorts	Ken Williams	80/3/09	SORT-BUBBLE.BS Program Description		86/5/09
Revisions to the Operating System	PolyLetter	82/1/03	SORT-HEAP.BS Program Description		86/5/09
Rhodes' Corollary to Hoare's Law	Murphy	88/6/07	SORT.GO bug correction	Abstract Systems	89/4/09
RIM for 8085	Bob Bybee	80/5/02	SORT.GO information	Ralph Kenyon	87/5/03
Robert E. Lee's Truce	Murphy	88/6/07	SORT.GO Program description	Ralph Kenyon	87/4/01
ROM 81 Bug	Ralph Kenyon	83/1/03	Sorting fast	Bill Sullivan	81/3/06
ROM Versions	PolyLetter	82/1/03	Sorts in Machine language Review	Ken Williams	80/3/09
RPN-CALCULATOR.BS Program Description		87/5/08	Source PolyUsers	PolyLetter	82/1/02
RS-232 cable connections	Ralph Kenyon	87/2/06	Space.AS Program Description		86/4/09
S-100 Board Bank	Bob Bybee	86/4/03	SPACE.AS Program Description		86/4/09
S-100 boards in the Poly	Bob Bybee	81/5/03	Speed Setting on Disk Drives	Ralph Kenyon	86/6/02
S-100 External Device Controller	Robert Johnson	86/5/03	Speedup keyboards	Frank Stearns	81/4/04
SA-400 Mini-Disk Drive Part 2	Frank Stearns	83/2/07	Spell.GO program description	Ralph Kenyon	87/3/01
Sattinger's Law	Murphy	88/6/07	Spell/(A:IS) comments	Frank Stearns	87/5/04
SAVEF	Ralph Kenyon	86/6/03	Spell/(A:IS) comments	Ralph Kenyon	87/5/05
SAVEP	Ralph Kenyon	86/6/03	Spelling Checker in Public Domain	Abstract Systems	89/4/10
SCAN.GO description	PolyLetter	83/2/06	SPELL 3.0 comments	Frank Stearns	87/4/02
SCOPY.GO program description	Chuck Thompson	81/6/05	SPELL 3.0 comments	Frank Stearns	87/5/04
Screen Layout form	Mark Sutherland	80/2/04	SPELL 3.0 comments	Ralph Kenyon	87/4/03
Screen Protection	Mark Sutherland	80/1/01	SPELL 3.0 comments	Ralph Kenyon	87/5/05
Screen redirected to Printer	Mark Sutherland	80/3/02	Spooler works with Exec/93	Bob Bybee	81/4/03
SCREEN-PROTECTED-DEMO.BS Program Description		86/5/09	Squeezing BASIC	Frank Stearns	81/2/09
SDIR.GO description	PolyLetter	82/4/06	Squeezing Space in BASIC	PolyLetter	82/4/08
Search.GO program description	PolyGlot	89/2/06	STAR-LANES.BS Program Description		86/5/09
SEAWAR.BS Program Description		86/5/09	STARS.BS Program Description		87/4/07
Select.GO program ad	Ralph Kenyon	81/4/05	Start Addresses Load Addresses and	PolyLetter	83/5/03
Select.GO program description	PolyGlot	89/2/06	START/STOP BASIC display	PolyLetter	81/3/03
SEP 84 Disk of Month description	PolyLetter	84/5/06	Static Electricity on 5" DSDD	Chuck Thompson	82/3/05
Serial Device Driver	Ralph Kenyon	88/1/03	Stereological analysis	Dr. Briarty	87/1/02
Service for Drives	ISR	89/6/01	Stereology Program description		87/2/08
SETNEW	Bob Bybee	86/4/03	Sticking Keys	Ralph Kenyon	87/1/07
SETNEW.GO program description	Chuck Thompson	81/6/04	STOCKS49.BS Program Description		87/5/08
Setting Disk Drive Speed	Ralph Kenyon	86/6/02	STOP Bug in BASIC	PolyLetter	82/1/05
Setup: How To Use	Charles Steinhauser	86/1/03	Stopping Scrolling displays	Russ Hobbs	81/5/06
Sex-Appeal.BS program description	Bob Bybee	81/6/05	Strip Blanks: BASIC Function	Charles Thompson	86/1/06
Shaw's Principle	Murphy	88/6/07	SUBMIT	Ralph Kenyon	87/6/02
Shugard SA-400	Frank Stearns	83/1/05	Submit.GO	Ralph Kenyon	88/1/02
Sideways Scrolling by Edit	Ralph Kenyon	87/2/02	Subscriber Address List: PolyLetter	Charles Steinhauser	86/1/04
SIM for 8085	Bob Bybee	80/5/02	SUN-RISE/SUN-SET.BS Program Description		86/5/09
SIN bug	Ralph Kenyon	88/3/08	SUNWAR.BS Program Description		87/4/07
SIN Bug in BASIC	Ralph Kenyon	83/3/07	Super Pilot	Tim Scully	81/4/02
Sio Input bug	Ralph Kenyon	83/3/07	Super Zip (for keyboards)	Frank Stearns	81/4/04
Sio.PS BugNote 17.0	Abstract Systems	89/1/09	SUPERMAN.BS		87/2/08
Sio.PS Program listing	Ralph Kenyon	88/1/05	Survey Form	PolyLetter	86/2/08
Skip Lines: BASIC Function	Charles Thompson	86/1/06	Survey Winners	Ralph Kenyon	86/3/08



Suspense file program (TICKLER.B5)	Ralph Kenyon	88/5/02
Syquest Bugs Fixed	PolyLetter	83/6/02
System Enhancements Exec/93	PolyLetter	81/1/01
System enhancements Exec/94	Don Moe	81/2/01
System Programmers Guide	Ralph Kenyon	86/4/06
System Programmers Guide (Msg)	Ralph Kenyon	88/3/09

---

### Advertising

Commercial advertising rates are \$50 for a full page, \$25 for a half page, and \$15 for a quarter page. Anything smaller is \$3.00 per column inch. A column is 3-3/4 inches wide by 10 inches tall. A full page is 7-5/8 inches wide. Noncommercial ads by subscribers are free.

---

PolyMorphic 8813 needs home. Make offer. Conway Spitler, P. O. Box 385, Fillmore, CA 93016-0385

---

Poly 8813, 3 drives, 64K, 2 printer ports, with a set of spare boards, 1 spare drive, 2 AJ-832 daisy-wheel letter quality printers, all documentation, etc. Any reasonable offer (must take all), Doug Schirripa, 716-724-5023 (days) or 716-657-7437 (evening).

---

Entire PolyMorphic System User Manual, System 88 User's Manual with Exec/96 addendum, & System 88 Operation Essentials On IBM disk. Al Levy, 516-293-8358

---

FOR SALE: PolyMorphic System 8813, 64K, three new double-sided drives (Not double density) (also reads & writes single-sided disks); high resolution PolyGrafix video card (640x240) w/inverse video capability (compatible with all standard graphics programs); Programmable Character Generator with software driver; 15" Sanyo monitor; GRI 777 keyboard; cassette & printer mini-cards; Integral Data Systems "BrighterWriter" printer. Enhanced system software. Excellent condition. Best offer. Ken Lowe, 5936 Zina Circle, West Valley City, UT 84120 (801) 969-7736.

---

FOR SALE: Poly 8810 box with power supply and mother board. \$50 plus shipping. Charles A. Thompson, 2909 Rosedale Avenue, Dallas, Texas 75205-1532, (214)-368-8223.

---

### DISKS - MODEMS - PROMS - SOFTWARE - SPELL \*\*\* New Low Prices \*\*\*

1. MAXELL diskettes: 5-1/4" 10 hard sector -- \$10 per box.
2. Used diskettes: 5-1/4" 10 hard sector -- \$0.50 each.
3. Hayes Micromodem 100 (300 baud S-100 internal modem) \$20.  
(If you don't have a modem this is a cheap way to go.)
4. HayesSys modem software (for the Micromodem 100) \$10.
5. Abstract Systems Exec (Enhancements & bugs corrected) \$30.
6. Abstract Systems Proms (Enhancements & bugs corrected) \$35.
7. PolyGlot Library Volumes: \$6 each; 5 or more - \$5 each.
8. Hayes Smartmodem 1200B (IBM compatible internal) \$40.  
Abstract Systems, etc., 191 White Oaks Road,  
Williamstown, MA 01267, Phone: (413) 458-3597  
(Send \$1.00 for a complete catalog--[free with any order].)  
(Make check or money order payable to Ralph Kenyon.)

---

### Service

Poly HD-18 controller cards, which was manufactured by Xebec, can be replaced or serviced by a subsidiary of Xebec, Omnishore, who is still in business at 1700 Forest Way, Carson City, NV 89706, Phone: (702) 883-8885.

Percy Roy reports that his Keyboard III is failing. The problems he describes sound like a key switch is wearing out. Keyboard III has individual switches for each key. Does anyone out there in PolyLand have a defunct Keyboard III that we could take apart for spare parts?

### Environment Corner

I've begun a new campaign to fight junk mail. I send it back to them in their own postage paid envelope. I include every scrap of junk they sent me, including the outer envelope. This way they not only have to dispose of it, they have to pay the return postage. I also send back the junk which comes with my various bills. If we all did this we would all probably get a lot less junk mail.

I have also made up a sticker which says:

These envelopes with cellophane or glassine windows cannot be recycled. Stop using them!

I attach the sticker to any offending envelopes in not only junk mail, but on any bills I pay.

**BugNote**

Abstract Systems BugNote 019.0 May 5, 1983

Edit 3.3 (6/10/81) has a bug in the exit routine. ESC CTRL-O RETURN closes the output file. If the output file has been closed and there is more data to be read in from the input file than will fit in memory, edit blows up on a normal exit sequence (ESC CTRL-E). Most of the time, my system reboots, but it has sometimes locked up. Only pushing the LOAD button works.

To avoid this problem, use ESC CTRL-X RETURN to exit when no output file is desired.

FORMAT will insert an extra space after abbreviations like Mr., Ms., Dr., etc. To prevent this extra space from being inserted, use the {point} command in place of the period. In the above line I have "Mr{point}, Ms{point}, Dr{point}, etc."

**In This Issue**

Editorial . . . . .	1
Letters . . . . .	1
Announcements PM 2.0. . . . .	1
PolyLetter: The first 10 Years . . . . .	2
Advertisements . . . . .	9
Service - Hard Disk Controller . . . . .	9
Environment Corner . . . . .	9
BugNote 19 - Exiting Edit . . . . .	10
The Bit Bucket . . . . .	10

**Bit Bucket**

Poly's WordMaster program FORMAT.GO inserts an extra space after periods, exclamation points, question marks, and other end of sentence punctuation. It checks the next to the last character for cases involving quotations, parenthesis, etc. One side effect of this is that

**Coming Soon**

Poly Meta, More: BASIC for Beginners, PC stuff, System Programmers Notes, Help, BugNotes, Public Domain Software, etc. More of PolyLetter: The first 10 years.

**PolyLetter**  
191 White Oaks Road  
Williamstown, MA 01267  
(413) 458-3597

Address Correction Requested

**FIRST CLASS MAIL**



Ralph E. Kenyon, Jr.	EXP:99#9
Abstract Systems, etc.	184
191 White Oaks Road	
Williamstown, MA	01267-2256


© Copyright 1990 by Ralph E. Kenyon, Jr.

PolyLetter Editor and Publisher: Ralph Kenyon. Subscriptions: US \$18.00 yr., Canada \$20.00 yr., Overseas \$25.00 yr., payable in US dollars to Ralph Kenyon. Editorial Contributions: Your contributions to this newsletter are always welcome. Articles, suggestions, for articles, or questions you'd like answered are readily accepted. This is your newsletter; please help support it. Non-commercial subscriber ads are free of charge. PolyLetter is not affiliated with PolyMorphic Systems.

Back volumes of PolyLetter (1980 thru 1989) are available at reduced prices payable in US dollars to Ralph Kenyon. 1 - \$15, 2 - \$28, 3 - \$40, 4 - \$50, 5 - \$59, 6 - \$67, 7 - \$75; Canada add \$3 shipping, Overseas add \$10. Individual back issues are also available (\$3.50, \$4.00, \$5.00).



# PolyLetter



PolyLetter 90/6

Page 1

NOV/DEC 1990

## *Editorial*

Our numbers have continued to dwindle. We are now down to 28 subscribers. I do, however, know of some Poly users who have no interest in PolyLetter. Their Poly's serves them quietly and without the benefit of our sharing.

When I assumed the helm in 1986 we were at slightly over 90 subscribers. I managed to build it up to over 100 for a while, but the last 5 years has seen a slow but steady decline. Because we are now an exclusive club of die-hards, we ought to communicate more. A few of us talk occasionally, but many are never heard from. Let's hear from the "silent majority". What would you like to see in PolyLetter? Take a bit of time and share your story with us. Send us letters; you don't HAVE to talk about your Poly.

## *Announcements*

Stanwood Associates, Inc., of Chicago submitted a review copy of SIMPLY INGENIOUS™ -- "The World's Simplest Database.

The program certainly lives up to it's name. Stanwood chose to name the main program "N.COM". It keeps a data file called notes.fil. Typing "N" and ENTER brings up the a screen with two simple instructions. Typing one or two words will cause Simply Ingenious to search through its database and display all records containing those words. Typing three or more words causes Simply Ingenious to tack on the date and time and store the text typed until ENTER is struck twice in a row. Each entry has the date and time of its creation. The main data file is stored in DOS TEXT format, so may be edited with any word processor which handles ASCII files. In the search and display mode records may be erased. Erasure procedes by menu selection and confirmattion. During the

actual erasure process, Simply Ingenious creates a notes.bak file; this prevents accidental data loss. Simply Ingenious is so simple to use its documentain fits on one two-sided sheet of paper, and even that is unnecessary. It boasts that most users are fully qualified in less than one minute!

An optional dialer is included; this allows one to dial using a modem direct from within the program. Also provided was a program to interface to d-Base III & d-Base IV files.

The program retails for \$49.95 and runs on DOS compatible computers. For further info contact: Stanwood Associates, Inc., 303 East Ohio Street, Chicago, IL 60611, (312) 828-9734 1-800-752-7982.

## *Letters*

Dear Ralph,

December 14, 1990

I've recently been asked to repair a couple of Poly hard disk drives. Not the ones I built, but the "real" 88/DS hard drive boxes that Poly used to sell. It's been interesting. As all good PolyLetter readers know, I built and sold hard drives to Poly users for several years, much to PolyMorphic's chagrin. But I had never seen one of their units, face to face, until recently. It's a nice box. Looks just like the old 8810 chassis, but with different cutouts on the front. And inside, it looks just about like the ones I used to build... even uses the same Xebec disk controller board.

This experience gave me two reasons to write to PolyLetter this month. First is the article about SASI/SCSI interfacing, which is included below. But it also gave me a great idea: why not do away with the external box, with its Xebec controller and power supply, and just put a small SCSI-interface drive right inside an 8813

chassis? It can take the place of one 5.25" floppy drive, using the same power supply connections. Imagine your 8813 with two floppies and a hard drive, all in one box! (Or three floppies, if you want to replace two of them with half-height drives.)

Today, SCSI drives can be had with as little as 20 MB of storage, or as much as 100 MB or more. Poly owners may want to consider this option if they have a hard drive that goes down, or if they just want to add more storage to an existing Poly system.

As they say on That's Incredible, "don't try this at home." This kind of change would require new software, since the Poly device drivers wouldn't know how to work with today's SCSI drives. Plus, the Poly host adapter card requires a minor change to keep it from damaging the new SCSI drives.

You may recall that PolyMorphic got quite upset at me for selling hard drives in the past. They felt I was in competition with them, and I agreed not to compete in that area any more. But now that Poly is officially out of business, I feel comfortable in declaring myself "back into" the hard drive business. If I can be of assistance to any of your readers, please have them contact me. -- Bob Bybee, Stone Mountain, Georgia.

[Bob, why did you ever agree not to compete with Poly? Competition is the life-blood of American business. Having a good, quality product at competitive prices benefits us all. Without competition, a company can price itself out of the market, as Poly appears to have done (among other problems). Please enlighten us; share your reasons for not competing with Poly. Bob can be reached at 404/498-3556 or Poly Peripherals, 5011 Brougham Court, Stone Mountain, GA 30087. -- Ed.]

Dear Ralph,

December 29, 1990

I am still struggling to make to HD's operational. Bob Bybee was unsuccessful in his attempt.

I would like to advertise in PolyLetter to buy an HD compatible with Poly.

Omnishore has control boards and makes

HD units, but they are unable to read the Poly system disk. They need specs on the HD unit -- specifically how many cylinders, sectors/cyl. and bytes/sector -- in order to supply the workable hardware. I have asked Bob Bybee for his input and would appreciate same from anyone else. Perhaps Parsei could help, but I do not have his current phone number or address?

My four HD's have found different hard drives, NEC, TULIN Corp., QUANTUM, and RODIME. I called Quantum who no longer make the model Q530 nor could they tell me who might have it.

So if you or anyone else can help, I would be grateful. Best wishes for the New Year, -- Jim Trahan, 2601 Section Road, Cincinnati, OH 45222, Phone: (513) 531-3106.

[Jim -- I seem to remember, among the documentation on disk that I got from Poly, that there were some modifications or jumpers to be set on the Xebec controller for operation on the Poly. As far as the hardware goes, I do know that the controller is set up for 256 byte sectors, 32 sectors per track using jumper W3. Unfortunately, I am presently unable to locate the Poly documentation. I will keep my eyes open for it.

What I might suggest is that you send your defunct controller to OmniShore and ask that they set up the replacement controller with the same modifications and/or jumpers as was on your old one.

In regard to the number of cylinders, heads, etc., that depends upon the drive. Poly's WCU program needs this information to properly format the drive. Many different drives are all compatible with Poly's Volume Manager driver software. I have some information needed to answer WCU's questions. These number are in DECIMAL and would need to be converted to HEX for the Poly WCU program. Here are the specs for some models of the drives you list:

Drive	Size	Hd	Cyl	RW	WP
NEC 5126	21	4	615	615	128
NEC 5146	44	8	615	615	128
Q520	18	4	512	256	256
Q530	27	6	512	256	256
Q540	36	8	512	256	256
TL226	22	4	640	999	300



TL240	33	6	640	999	300
RO201	6	2	320	132	0
RO201E	11	2	640	132	0
RO202	11	4	320	132	0
RO202E	22	4	640	132	0
RO203	16	6	320	132	0
RO203E	33	6	640	132	0
RO204	22	8	320	132	0
RO204E	44	8	640	132	0
RO252	10	4	306	80	80
RO351	5	2	306	80	80
RO352	10	4	306	80	80

Cap = Capacity in megabytes

Hd = Number of heads

RW = Reduced Write Cylinder

WP = Write Precompensation Cylinder

The parking cylinder should be larger than the total number by a number of tracks. I don't know what's best, but you could just add 10 or 20 to the list. The drive will try to step out how much you say, but will stop at its physical limit.

I have the specs on numerous other drives, but my list is not complete. Any drive which uses the ST506/ST412 interface should work. I am using an IBMWD25 in mine. Any full height or half height drive which supports MFM format should work provided it uses the ST506/ST412 interface. If you have any particular drive you'd like info on, I'd be happy to look it up in my list of 183 drives.

OmniShore won't be able to format a drive for you. You will need to do that using Poly's WCU.GO. Oh, they may be able to format it to Poly's specifications, but WCU.GO wants to write the parameters on the drive for Driver.DD to access, and I don't know if the checksum for that data will be computed in the same way. -- Ed.]

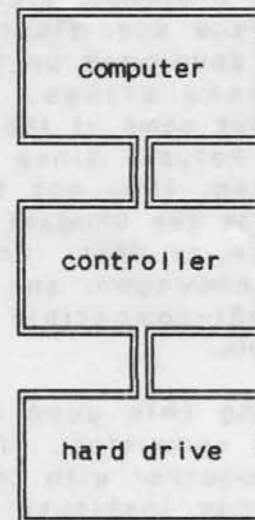
### **What's Scuzzy about SCSI?**

by Bob Bybee

Hard disk storage has come a long way since the early 1980's. When I first started working with hard drives, the "state of the art" was a 10 megabyte drive with 8-inch platters, that weighed about 40 pounds and cost over \$1000. Plus, you needed a fancy disk controller to talk to it. The controller itself cost \$700. No wonder we "pioneers" paid an arm and a leg for our first hard drives!

The \$700 disk controller I'm referring

to was a very complex, high-speed piece of electronics. It had to be, 'cause it had to keep up with a fast hard drive that was slinging about 5 million bits per second at it. Our little computers weren't nearly capable of keeping up with that kind of data, so the controller was kind enough to sit between us and that nasty, fire-breathing disk drive. Here's how the computer, the controller and the drive all fit together.



Fine with us! We didn't want to have our poor little 8080 processor dealing with that hard drive directly anyway. Bad enough that our CPU has to keep up with tracks, cylinders, sectors and bytes on our floppy disks. We were happy to let this other controller handle the details as far as talking to the hard disk.

The controller simplified our view of the hard disk. Instead of seeing it as tracks, sectors, and so on, we could now think of the drive as just a series of "logical blocks." In fact, that's just how Dio, the Poly ROM entry point for disk I/O, wanted to see things anyway. So this controller was a perfect match for Dio.

There was a defined protocol, or language, for talking to this controller. You sent it a "command block" of 6 bytes, which specified things like:

- Which drive? There could be several, but often there was only one hard drive attached to the controller. Some of the controllers even let you add floppies or tape drives, though Poly never used that feature.

- What to do? Typically it would be a read, write, or format command.

- Where to start? This was a block number, from 0 to the maximum number of blocks available on the drive.

- How many blocks to do? Another number.

If you've ever written a program on the Poly that calls Dio, these items should look familiar. The parameters you pass to Dio are exactly these four items, passed in the A, B, C, and HL registers. (Dio also requires the memory address in DE.)

This command protocol, along with the electrical interface specification for the controller, was developed by Shugart for use on their hard drives. Remember Shugart? They put some of the first 5.25" floppies in our Polys. Since they wrote this specification, they got to name it, and they called it the Shugart Associates Standard Interface, or SASI. Other vendors hopped on the bandwagon, and soon there were a dozen SASI-compatible drives and controllers available.

Well, anything this good has to get screwed up by a committee. The vendors decided to get together with the American National Standards Institute (ANSI) and cast this specification in concrete. The SASI spec was developed mainly for hard drives, but they could see that it might come in handy for other devices, like tape drives, optical read-only or write-once drives, or even printers. So they added. And they extended. And they renamed. The old SASI standard became the new, improved, committee-ized Small Computer Systems Interface, or SCSI. Most people pronounce it "scuzzy."

Actually, SCSI came out fairly well, compared to most committee outputs. It works, it has some nice enhancements over SASI, and it even retains compatibility with SASI in most areas. One problem with SASI was that different equipment, from different vendors, handled certain things in vendor-dependent ways. Disk errors, for example. So, you either had to stick with one vendor's method of doing things, and not buy drives/controllers from other vendors, or you had to make your software very smart, or ship different versions of software depending on whose disk controller you bought this month. Not fun. SCSI did away with most of those problems. But it is also different enough from SASI, that your old SASI hardware and software won't "plug in and play" with SCSI equipment.

So what's so great about SCSI? In addition to the simple command protocol, SCSI has a simple electrical protocol for connecting the computer to the controller. It sends 8 bits (one byte) at a time. The controller always asks for what it wants next, which keeps things simple for the CPU side. And the method of moving each byte across the interface is short and sweet:

1. The controller says REQUEST. I want a transfer.
2. The CPU reads a byte, or writes a byte, depending on which direction the data is flowing for that transfer.
3. The CPU says ACKNOWLEDGE. I'm done with your request. Next byte!

This is so simple, it takes only two wires: one for the request signal and one for the ack. There are also 8 data lines, plus a couple of extra control signals that are used only during the start and end of the transfer, and there's an optional parity bit to check on data accuracy. In total, SCSI requires about 18 wires plus ground wires.

With an interface this simple, you'd expect it to be high-speed. And it is. SCSI specifies that you can move 1 megabyte per second. Most disk drives can't keep up with that rate, so the limiting factor is the drive, not the SCSI interface. There are optional features in the SCSI spec that can push the data rate up to 4 MB/second, but those aren't used in most applications.

#### Imbedded SCSI

The "old way" was to have your CPU talk to a host adapter board in your computer. The host adapter talked to the SCSI controller, and the SCSI controller talked to the drive. What if the SCSI controller were built into the hard drive? Now they are. It's hard to even find a separate SCSI controller anymore. (Xebec went out of business.) To put a SCSI drive into an IBM-PC, all you need is a "host adapter" card that goes into an IBM card slot, a drive, and some software (usually provided with the host adapter).

Do all PCs use SCSI drives? No. Why not? Early PC software was written especially for a different type of controller, so there's an historical bias against SCSI in the PC market. But this other controller can only handle drives up



to a certain capacity, whereas SCSI drives can be built with virtually unlimited capacity. There's one available now with over a GIGabyte on it.

Non-PC systems, such as UNIX workstations, don't have this historical bias. They also tend to have a larger appetite for disk space than MS-DOS based machines, so it's common to see SCSI drives in larger computers.

#### Smart Drives

One nice feature about SCSI is that the drives are very smart. If the drive media has any "bad blocks," these are detected and marked when the drive is formatted, so the drive always appears defect-free once it's installed. Some SCSI drives even detect bad blocks which "grow" during normal operation as the drive ages, and re-map these blocks too, so it's very rare for a disk error to occur on a modern SCSI drive.

But my favorite SCSI feature is the one called "inquiry." Software in the host computer can ask the drive for its capacity (number of blocks), its organization in tracks, sectors, and heads, the name of its manufacturer and the drive model number, and probably other information that nobody would ever care to know. So the host software never needs to be configured for a particular drive; it can be made self-configuring, so it can work with drives of all sizes.

#### Other Interfaces

I'll wind up this article with a quick overview of the other types of disk interfaces commonly available today. Each of these uses a different type of controller and/or host adapter, and the software for each one is different. Compatibility is but a dream.

**ST-506.** Seagate Technology created this for their early 5.25" hard drives. ST-412 is a similar interface. The controller card is quite complex since the drive doesn't have much "smarts" on it. Probably more than 90% of all PCs have ST-506 controller cards and drives.

**ESDI: Enhanced Small Disk Interface.** Used for large, fast drives, usually even larger and faster than SCSI. Commonly used

in workstations and file servers. But again, the controller is complex and the drives aren't very smart.

**IDE: Integrated Drive Electronics.** This is one of the newer standards for drive interfacing. As the name implies, this one puts most of the disk-controller smarts into the drive, rather than into the host computer, so the interface card in the computer becomes quite simple and cheap. Found in many new AT-type systems.

**IPI and IPI-2: Intelligent Peripheral Interface.** This one is fairly rare, but it provides a "smart" interface like SCSI.

**SCSI-2: A new enhanced version of SCSI.** Has 16-bit transfers for more speed, and a number of other new features.

[Editor's note. Just to keep apples and oranges separate, Shugart Associates System Interface (SASI) describes the way the computer must communicate with the controller. The host adaptor makes this physically possible. Poly's host adaptor is just a set of glorified parallel status and data ports connecting the S-100 bus to the right sized cable. The Shugart Technology ST506 interface specification describes the electrical and timing interface to the drive. It's actually quite similar to the interface between the Poly's disk controller and the SA-400 drives, but with a few more control lines and requiring faster and more precise timing. Poly's double density controllers are "intelligent" controllers and perform many of the functions that SASI or SCSI controllers do for hard drives. As Bob mentioned, SCSI "drives" are really drives with the controller built in, and the SCSI interface specification is different from the SASI.

Poly's Winchester Controller Utility program, WCU.GO, detects bad sectors when the drive is formatted, and the device driver code in Driver.DD performs the same service during the normal course of operating the drive. Poly's Volume Manager software maintains a 32 sector pool (one track) on any hard drive for replacements for bad sectors; these sectors are automatically allocated as replacements for bad sectors discovered during operation -- "on the fly". Poly users will normally not see any hard disk errors; the automatic allocation process reassigns sectors from

the pool as the bad sectors are detected. The device driver code rewrites the device configuration block to record the location of bad sectors as they accumulate. When you use WCU to park the hard disk, it will report the number of bad sectors that have accumulated to date. If that number is "20" (32 decimal), then the pool is used up.

Bob mentioned that "Some SCSI drives even detect bad blocks which 'grow' during normal operation as the drive ages, and re-map these blocks..." This is a case of putting not only the controller on the drive, but some of the supporting device driver software as well.

The "inquiry" feature Bob spoke of is only available because the data is written on the drive during it's initial formatting. Poly's WCU program writes whatever you tell it as the drive configuration when the drive is formatted. That usually includes the formatting information and the manufacture. Of course, that information must be available when it comes time to configure the drive ("partition" in pc parlance). Poly's CONFIGURE.GO program "enquires" of the drive its tracks and configuration by reading it from the drive. -- Ed.]

### *Edit.GO*

One of the things that Edit.GO does when you exit is to "erase" each of the block markers you may have put into the file but didn't take time to delete. However, sometime in the many versions Poly produced, this got to be a slow process. Poly took steps to speed the process up, but that resulted in another anomaly. When Edit "erases" a block marker, it just converts it into an ascii null (00), but leaves it in the file. Edit gets rid of these during the load process when starting to EDit the file again. It removes any nulls. This works fine most of the time, but lately, I have been capturing the output of format.GO in a file and hand editing a few characters. Unfortunately, when a graphics character contains nulls, Edit strips them out too. (I found this out when I created the Copywrite symbol using graphics.)

### *Geneology*

Lately, I have become interested in

geneology issues. At a family reunion one cousin suggested that I put all the relatives (personal data, not persons) on my computer. I began by just collecting names and addresses, but I soon added relationship data to keep track of who was related to whom and how. I began creating a program to browse through the data base using relationship links.

Some geneology information pertains to one person -- name, address, telephone number, etc. Some geneology data pertains to two persons -- who gave birth to whom and when, who married whom and when, who widowed whom and when, etc. After much consideration I settled on two source files of data and three compiled files.

I put one-person information in a file called "FAMILY-NAMES.TX". This file has the person's name, nick-name, address, phone number, and sex. I put two-person information in a file called "FAMILY-EVENTS.TX". This file has the names of the two persons, the date, and a code letter for the event. I have codes for the following events. B - gives birth to, A - adopts by mother, a - adopts by father, F - fathers, L - lives with, M - marries, S - separates, D - divorce, W - is widowed by, and Z - is survived by.

Both the FAMILY-NAMES and FAMILY-EVENTS files are hand edited files but have particular columns for the data. But, since BASIC runs very slow on variable length record files, I have programs which compile both these files into fixed length record files. COMPILE-NAMES simply makes a fixed length data file with the names from FAMILY-NAMES.TX. COMPILE-EVENTS is, however, somewhat more complicated. It reads the names into an array and sorts them. Each name in the EVENTS file is looked up (with a binary search); the NAMES file record number is used in the EVENTS.DT file rather than the actual name text.

I have a third program, COMPILE-VECTORS, which creates a data file with links from a person to certain other relatives. The links for a person include mother, father, spouse, older sibling, younger sibling, and youngest child. By compiling this data in a file it is possible to move around the data base very quickly. I use Don Moe's Form.OV program to display a person's data and a linked event on one screen. By pressing various keys, I can skip, in the

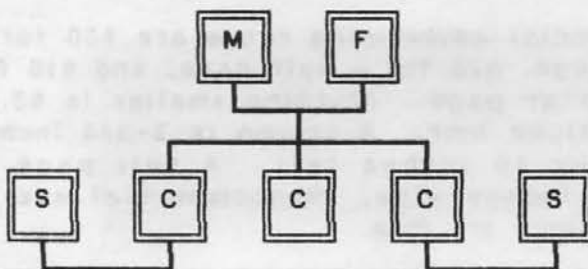


blink of a disk access, to any one of the relations linked in the vector file. Compiling the .TX files into the .DT files takes a bit of time; but once compilation is complete, access to the relations is quite fast.

The final program allows printing a couple of kinds of reports. One is the geneology data for a person. This printout is basically a data collection and verification form that lists all data on one person -- including all relations. A second report is a family tree of the descendents of a selected person. My original idea was to present my 86 year old maternal grandmother with a complete list of all her descendents and their spouses. But it mushroomed into something bigger.

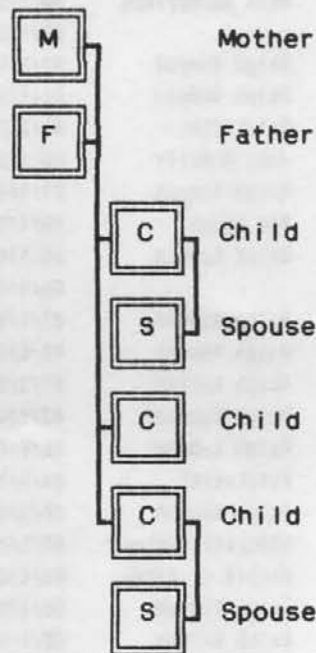
Family trees are usually presented horizontally. Computers don't like horizontal trees. To make my computer happy, I turned the tree on its side. The ancestors of a person get printed first -- the branches of the tree. The descendents of the person get printed second -- the roots of the tree. Right now the system can handle up to 6 generations.

The task has presented some interesting BASIC programming problems. For example, printing a list of a person's descendents required thinking about how to arrange the list and how to program the computer to follow the design. On a horizontal tree, a couple's children are lined up, side by side, under the couple. A person's spouse is linked to one side. Here's what it looks like:



Turning this on its side in such a way that there is space to write a person's name to the right of the box requires shifting things around a bit. Instead of putting the children just to the right of the parents, I move them down. But, I also always put a spouse below the blood family member. Vertical lines link parents with

children. Parents are on the left, children on the right.



10 Year Index (part 3)

System Revisions	PolyLetter	82/1/03
SYSTEM-SORT-PRINT.BS Program Description		86/5/09
Szap.GO program use	Bob Bybee	81/6/06
T-TEST.BS Program Description		87/5/08
TABBER.BS Description	PolyLetter	81/2/08
Table-Driven Programs	Bob Bybee	84/2/02
Take Heart Dear PolyOwner	Al Levy	85/2/04
TAPE-LOAD.AS Program Description		86/4/09
TELETYPE-CODE.BS Program Description		86/5/09
Terminal programs - PGL-V-12		87/1/08
TERMINAL-FC.AS Program Description		86/4/09
TEXT-TRAN.BS description	PolyLetter	82/4/06
The "System" Disk	PolyLetter	82/5/06
The Bird Has a Name	PolyLetter	84/4/01
The C. A. Thompson Method	Charles Thompson	80/5/02
The Central Processing Unit (CPU)	Bob Bybee	84/6/02
The Essence of VaporWear	Charles Steinhauser	87/4/06
The First Bug	Ralph Kenyon	86/5/05
The Last One's Law of Program Generations	Murphy	88/6/07
The New System	Ralph Kenyon	86/4/04
The Other BASIC	Ralph Kenyon	88/2/09
The Other BASIC	Ralph Kenyon	88/3/07
The Other Guys	Ralph Kenyon	86/4/03
The Other Guys	Ralph Kenyon	86/6/09
The Other Guys (IBM)	Ralph Kenyon	87/3/08
The Programmers Calculator	Bob Bybee	80/3/02
The Rescue	PolyLetter	84/1/04
The Stack	PolyLetter	82/6/02
The Straight Wire	Ralph Kenyon	87/2/06
The Top of Memory Is....	John Warkentin	80/2/05
The Ultimate Program	Bob Bybee	80/1/06
Thoreau's Theories of Adaptation	Murphy	88/6/07
TI 820 Printer Interface	Bob Bybee	80/5/06

TICKLER.BS	Ralph Kenyon	88/5/02	WH1 - Assembly Language Output	Ralph Kenyon	86/4/06
Time Clock	Dan DeForest	80/3/08	What Do You Say to a Dead Poly?	PolyLetter	84/2/03
Time Display as Hours and Minutes	Al Levy	86/1/07	What Do You Say to a Dead Poly?	PolyLetter	84/3/01
Time Sharing for PolyLetter	Mark Sutherland	80/3/01	What else don't we need?	PolyLetter	82/3/04
TIMEBOMB.BS Program Description		87/4/07	What to do with a New Disk	Al Levy	87/4/09
TITLE on PAGE command	Ralph Kenyon	80/3/09	What's Copyrightable Anyway	Ralph Kenyon	89/2/02
Today.AS source code	Ralph Kenyon	81/1/07	What's in a ROM	Bob Bybee	84/3/05
Tran.OV Description	PolyLetter	81/2/08	What's New at PolyMorphic	Bob Bybee	83/4/01
Trouble Using a Plotter	John McNally	86/4/01	Who Needs Vmgr, Anyway	PolyLetter	82/3/04
True Circle -- User profile	Ralph Kenyon	89/4/03	Why a PolyLetter	Mark Sutherland	80/1/01
TSR Programs	Bob Bybee	88/4/07	Why an HD/18	PolyLetter	83/2/05
TSR programs for the Poly	Ralph Kenyon	88/5/05	Why does it cost so much?	Bob Bybee	83/4/01
Tweak.AS Program Description		86/4/09	Why Review IBM software	Ralph Kenyon	89/1/01
Twin System	Ralph Kenyon	87/1/02	Windows for the Poly		87/2/08
Two Poly TSR programs.	Ralph Kenyon	88/5/05	Winners: Survey	Ralph Kenyon	86/3/08
Two's complement	Ralph Kenyon	81/3/08	Wood's Axiom	Murphy	88/6/08
TYPE Bug	Ralph Kenyon	83/1/03	Wooden Keyboard		87/5/09
TYPE on a Clone	Ralph Kenyon	86/4/03	WordMaster II	Mark Sutherland	80/4/05
Typesetting with your Poly	PolyLetter	84/6/01	WordMaster II and Keyboards	Ralph Kenyon	86/3/05
UCSD PASCAL	Ralph Kenyon	89/3/01	WordMaster II hint	Peter Jackson	81/4/03
UNDELETE COMMAND HELP	Abstract Systems	89/6/09	WordMaster II Index	Charles Thompson	81/2/06
Under the Double Integral	Harold C. Kinne	80/4/03	WordMaster Released	Mark Sutherland	80/4/01
UNSAVEP (part 1)	Ralph Kenyon	88/2/06	WordMaster Tricks	PolyLetter	84/4/02
UNSAVEP (part 2)	Ralph Kenyon	88/3/07	Wormhole Driver Description	Ralph Kenyon	86/5/06
UNSAVEP (part 3)	Ralph Kenyon	88/4/09	WRITE Bug in BASIC	Ralph Kenyon	83/1/03
UNSAVEP (part 4)	Ralph Kenyon	88/5/02	WRITE Bug in BASIC C03 using MAT	Ralph Kenyon	82/1/06
Upgrading an 8810 to Two Drives	PolyLetter	83/4/05	Yes or No BASIC function	Jonathan Wolfret	81/3/06
User Defined Functions in BASIC	Charles Thompson	82/1/08	Your wish is Poly's Command (file)	PolyLetter	84/1/03
User Defined Functions: BASIC	Charles Thompson	86/1/06	YPS-42.BS program description		87/2/08
User Input Program Review	Mark Sutherland	80/1/05	Z-80 Cross assembler comments	Ron Moffatt	87/4/04
User profile - True Circle	Ralph Kenyon	89/4/03	Z80 Cross Assembler		87/1/08
User Report on Stuart Woods	Mark Sutherland	80/2/02	ZAP HELP	Ralph Kenyon	88/3/09
Users Address List	PolyLetter	89/1/02	Zits!	Ralph Kenyon	86/6/01
Users Group for PolyMorphic		80/1/02	Zymurgy's First Law of Evolving System Dynamics	Murphy	88/6/08
Using Exec from BASIC	Bob Bybee	80/5/04	(cpi n)	Chuck Thompson	83/4/09
Using FORMAT from Exec	Russ Hubbs	82/4/09	(lpi n)	Chuck Thompson	83/4/09
Using Format from Exec; More	Russ Hobbs	82/5/04	(nxt ...)	Chuck Thompson	81/3/04
Using FTP as a terminal	PolyLetter	83/5/02	(tabs n)	Chuck Thompson	83/4/09
Using Mailist - Part 4	Al Levy	86/1/02			
Using Super Zap	Bob Bybee	81/6/06			
Using Two Printers from BASIC	Chuck Thompson	83/4/09			
Using WordMaster II Without WPS	PolyLetter	82/3/07			
Using Your IBM Color Monitor	Don Haywood	88/2/01			
Virtual Universe Operating System	PolyLetter	81/4/06			
VOL.GO program description	PolyGlot	89/2/07			
Volume Manager Explained	John Warkentin	83/1/04			
VSelect.GO program ad	Ralph Kenyon	81/4/05			
VSelect.GO program description	PolyGlot	89/2/07			
Wait HELP	Ralph Kenyon	88/3/09			
Wait States	Ralph Kenyon	89/3/03			
Wait.AS Program Description		86/4/09			
Wanted: CASHFLOW	Ralph Kenyon	86/6/02			
Wanted: Checkbook Program	Charles Mach	86/6/01			
Wanted: Financial Programs	James Salinger	86/4/09			
Wanted: Inventory System	Travis Miller	86/3/05			
Wanted: Pascal	Percy Roy	86/5/01			
Warm Starting (Bomb Shelter)	Mark Sutherland	80/3/06			
Weinberg's Corollary	Murphy	88/6/08			
Weinberg's Law	Murphy	88/6/08			
WHO - Assembly Language Input	Ralph Kenyon	86/4/06			

---

### Advertising

Commercial advertising rates are \$50 for a full page, \$25 for a half page, and \$15 for a quarter page. Anything smaller is \$3.00 per column inch. A column is 3-3/4 inches wide by 10 inches tall. A full page is 7-5/8 inches wide. Noncommercial ads by subscribers are free.

---

PolyMorphic 8813 needs home. Make offer. Conway Spitler, P. O. Box 385, Fillmore, CA 93016-0385

---

WANTED TO BUY: Poly Hard Disk. -- James Trahan, 2601 Section Road, Cincinnati, OH 45222, Phone: (513) 531-3106.

---



Poly 8813, 3 drives, 64K, 2 printer ports, with a set of spare boards, 1 spare drive, 2 AJ-832 daisy-wheel letter quality printers, all documentation, etc. Any reasonable offer (must take all), Doug Schirripa, 716-724-5023 (days) or 716-657-7437 (evening).

Entire PolyMorphic System User Manual, System 88 User's Manual with Exec/96 addendum, & System 88 Operation Essentials On IBM disk. Al Levy, 516-293-8358

FOR SALE: PolyMorphic System 8813, 64K, three new double-sided drives (Not double density) (also reads & writes single-sided disks); high resolution PolyGrafix video card (640x240) w/inverse video capability (compatible with all standard graphics programs); Programmable Character Generator with software driver; 15" Sanyo monitor; GRI 777 keyboard; cassette & printer mini-cards; Integral Data Systems "BrighterWriter" printer. Enhanced system software. Excellent condition. Best offer. Ken Lowe, 5936 Zina Circle, West Valley City, UT 84120 (801) 969-7736.

FOR SALE: Poly 8810 box with power supply and mother board. \$50 plus shipping. Charles A. Thompson, 2909 Rosedale Avenue, Dallas, Texas 75205-1532, (214)-368-8223.

**DISKS - MODEMS - PROMS - SOFTWARE - SPELL**  
\*\*\* New Low Prices \*\*\*

1. MAXELL diskettes: 5-1/4" 10 hard sector -- \$10 per box.
2. Used diskettes: 5-1/4" 10 hard sector -- \$0.50 each.
3. Hayes Micromodem 100 (300 baud S-100 internal modem) \$20. (If you don't have a modem this is a cheap way to go.)
4. HayesSys modem software (for the Micromodem 100) \$10.
5. Abstract Systems Exec (Enhancements & bugs corrected) \$30.
6. Abstract Systems Proms (Enhancements & bugs corrected) \$35.
7. PolyGlut Library Volumes: \$6 each; 5 or more - \$5 each.
8. Hayes Smartmodem 1200B (IBM compatible internal) \$40.  
Abstract Systems, etc., 191 White Oaks Road,  
Williamstown, MA 01267, Phone: (413) 458-3597  
(Send \$1.00 for a complete catalog--free with any order.)  
(Make check or money order payable to Ralph Kenyon.)

**Annual Index - 1990**

Abstract Systems Exec information	Ralph Kenyon	90/2/07
Basic Pointer	Ralph Kenyon	90/4/03
BASIC SAVE, and SAVE; - Bit Bucket	Ralph Kenyon	90/4/10

Benchmarks for the Poly Emulator	Bob Bybee	90/1/02
BugNote 14 - Sio loses characters	Abstract Systems	90/1/09
Cellophane - Environment Corner	Ralph Kenyon	90/5/09
COMP-DISK	Ralph Kenyon	90/1/10
DIRCOPY.GO operation	Ralph Kenyon	90/2/06
Directory Commands	Ralph Kenyon	90/1/07
Disk Maintenance Commands	Ralph Kenyon	90/1/08
Double column printing	Ralph Kenyon	90/2/08
Edit and ^^ - The Bit Bucket	Ralph Kenyon	90/6/10
Edit.GO - Erasing block markers	Ralph Kenyon	90/6/06
Editorial	Ralph Kenyon	90/1/01
Editorial	Ralph Kenyon	90/2/01
Editorial	Ralph Kenyon	90/3/01
Editorial	Ralph Kenyon	90/4/01
Editorial	Ralph Kenyon	90/5/01
Editorial	Ralph Kenyon	90/6/01
Evaluating Benchmarks	Ralph Kenyon	90/1/03
File Handling Commands	Ralph Kenyon	90/1/08
For Beginners	Ralph Kenyon	90/1/07
Geneology	Ralph Kenyon	90/6/06
HELP COMMAND TYPE	Abstract Systems	90/1/07
HELP COMMAND UNDELETE	Abstract Systems	90/1/07
Horse Sense - Bit Bucket	Ralph Kenyon	90/3/09
Italics	Ralph Kenyon	90/2/09
Keyboard Processing Commands	Ralph Kenyon	90/1/08
Letter	Bob Bybee	90/4/01
Letter	Bob Bybee	90/6/01
Letter	James Salinger	90/3/02
Letter	James Salinger	90/5/01
Letter	Jim Trahan	90/6/02
Letter	Karl Thomas	90/3/02
Letter	Richard Wagner	90/2/01
Letter	Russ Nobbs	90/1/01
Letter to Quaker Oats	Ralph Kenyon	90/1/01
Managing TSR programs - part 2	Ralph Kenyon	90/3/01
Managing TSR programs on the Poly	Ralph Kenyon	90/3/07
MICROMODEM-100	Ralph Kenyon	90/2/09
Modem Programs	Ralph Kenyon	90/2/09
Octobit - Poem - Bit Bucket	Ralph Kenyon	90/1/10
PACK-CRASH	Ralph Kenyon	90/3/03
PC Outline review	Ralph Kenyon	90/3/06
PGL-V-33 - 10 year index to PL	PolyLetter	90/4/04
PM Bug - Poly Emulator Bug	Ralph Kenyon	90/1/04
Poly Emulator 1.3 Announcement	Bob Bybee	90/1/01
Poly Emulator 2.0 Announcement	Bob Bybee	90/5/01
Poly PROMS - and hardware versions	Ralph Kenyon	90/2/04
PolyLetter's 1st 10 Yrs. Part 1.	PolyLetter	90/4/05
PolyLetter's 1st 10 Yrs. Part 2.	PolyLetter	90/5/02
PolyLetter's 1st 10 Yrs. Part 3.	PolyLetter	90/6/07
Primitive Displays	Ralph Kenyon	90/1/06
Printer Control Commands	Ralph Kenyon	90/1/08
Printer Sharing	Ralph Kenyon	90/1/05
Program Control Commands	Ralph Kenyon	90/1/08
Recycling	Ralph Kenyon	90/3/01
Serial vs Parallel	Bob Bybee	90/4/03
Service: Xebec, Keyboard need	PolyLetter	90/5/09
Simply Ingenious	PolyLetter	90/6/01
slist.GO	Ralph Kenyon	90/3/02
System Functioning Commands	Ralph Kenyon	90/1/08
The Other Guys PLOT - LOCATE	Ralph Kenyon	90/3/06

Volume Manager Operation	Ralph Kenyon	90/2/05
What Scuzzy about SCSI?	Bob Bybee	90/6/03
WordMaster Punctuation - Bit bucket	Ralph Kenyon	90/5/10

**Bit Bucket**

When Edit saves an ESCape definition library, it uses the carrot character as a control character prefix. ^[ is ESC; ^A is CTRL-A. But, in order to get an actual carrot character, one must double it. ^^ becomes ^. The one place this doesn't work quite properly is when you try to attach an escape sequence to ^ itself. Edit won't save such definitions correctly. To fix this, you should EDit TXdef.ED and insert an extra ^^ after the definition. Before: ^[=^TEST^[; after: ^[=^^TEST^[.

According to a recent ABC News feature item, "Grunts" call Combat Boots "LPC's" (Leather Personnel Carriers).

**In This Issue**

Editorial	1
Announcements - Simply Ingenious	1
Letters	1
What Scuzzy about SCSI?	3
Edit.GO - Block markers	6
Geneology	6
PolyLetter: The first 10 Years	7
Advertisements	8
Annual Index - 1990	9
The Bit Bucket	10

**Coming Soon**

Poly Meta, More: BASIC for Beginners, PC stuff, System Programmers Notes, Help, BugNotes, Public Domain Software, etc.

**PolyLetter**  
 191 White Oaks Road  
 Williamstown, MA 01267  
 (413) 458-3597

Address Correction Requested

**FIRST CLASS MAIL**



Ralph E. Kenyon, Jr.                      EXP:99#9  
 Abstract Systems, etc.                      184  
 191 White Oaks Road  
 Williamstown, MA                              01267-2256

© Copyright 1990 by Ralph E. Kenyon, Jr.

PolyLetter Editor and Publisher: Ralph Kenyon. Subscriptions: US \$18.00 yr., Canada \$20.00 yr., Overseas \$25.00 yr., payable in US dollars to Ralph Kenyon. Editorial Contributions: Your contributions to this newsletter are always welcome. Articles, suggestions, for articles, or questions you'd like answered are readily accepted. This is your newsletter; please help support it. Non-commercial subscriber ads are free of charge. PolyLetter is not affiliated with PolyMorphic Systems.

Back volumes of *PolyLetter* (1980 thru 1989) are available at reduced prices payable in US dollars to Ralph Kenyon. 1 - \$15, 2 - \$28, 3 - \$40, 4 - \$50, 5 - \$59, 6 - \$67, 7 - \$75; Canada add \$3 shipping, Overseas add \$10. Individual back issues are also available (\$3.50, \$4.00, \$5.00).