

# THE GURU'S LAIR

Don Lancaster

## PostScript™ Flutterwumpers

Pick a peck of PostScript PIC's.

With a proper "flutfile" partitioning, all the raw power of real PostScript can easily get applied to stunningly low cost PIC robotic systems.



I have long been a rabid fan of that incredibly superb and totally general purpose PostScript computing language. For years, I have chosen PostScript as my first and foremost weapon for nearly all of my columns and stories, all of my primary engineering designs, and all of my technical illustrations. Plus bunches of other stuff ranging from investment models to my new magic sinewave research to Smith Charts to shaft encoders to web site traffic analyzers to hot tub controllers.

And, as we shall shortly see, to explore some utterly incredible new low end robotics opportunities.

Until recently, you really had to want to use PostScript-as-language in the worst sort of way to be able to do so. In particular, needing a PostScript printer to serve as the host computer was horribly limiting. The PostScript hard drives were somewhat few and far between. As was the lack of a real time visual display. And even more brutally crippling was that one-way parallel port that nearly all early PC systems had irrationally and stupidly forced upon PostScript users.

But these limitations are ancient history. PostScript is now literally busting out all over with brand new

capabilities and unique powers.

For instance, the free [GhostScript 5.01](#) clone gives you full host based PostScript computing combined with full screen visuals. GhostScript has gotten a lot better lately.

Variations on GhostScript can be customized for new commercial apps. As an example, the [Videonics](#) people now offer the *PowerScript 1000* fully animated video character generator. Thus turning zillions of fonts and all of the usual desktop publishing tools loose on video editing.

Original PostScript lacked most transparency features. The (literally) brilliant Videonics workaround was to redefine their CMYK color space, with "K" now serving as a new alpha transparency channel.

Host-to-printer interactions have dramatically improved. With higher speed two-way parallel comm newly added to the readily available options of Serial, AppleTalk, and Ethernet. Program apps such as that [Download Mechanic](#) from *Acquired Knowledge* simplify PostScript-as-language and printer interactions. With freebie eval demos available.

Meanwhile, [Adobe](#) has improved their *Acrobat 3.01* PostScript variant. Acrobat very greatly revolutionizes the distribution of information. Both off and on line. I do feel that Acrobat completely blows HTML away on all counts. Ludicrously so. By offering an exact control of what the end user will see, with *full screen* magnifiable displays *while still online*, and vastly improved antialiased visibility.

Combined with *url* hot linking, interactive forms, wipes, full text or entire library searching, bookmarks, thumbnails, byte range loading, scan capture, plus powerful handicapped access provisions. At typical single file sizes of 11K per fancy illustrated page. While printing equally well to either PostScript or certain newer non-PostScript printers.

One element of Acrobat is called the Adobe *Acrobat Distiller 3.01*.

Distiller's intended purpose is to let you convert complex PostScript code into simpler Acrobat .PDF files. But don't let the name fool you. *The Acrobat Distiller is really a speedy*

*genuine Adobe host based general purpose PostScript computer!*

One that makes PostScript fast and fun to play with.

## ACROBAT DISTILLER AS A POSTSCRIPT COMPUTER

A PC, UNIX, or Macintosh based *Acrobat Distiller 3.01* makes a fairly powerful general purpose PostScript computer. At its input, the Distiller accepts PostScript code from a disk text file. Distiller can also easily be taught ahead of time *to read nearly any file format in any language!* At the output, Distiller can generate its (usually intended) .PDF file. Or could write messages or calculated results to its (usually ignored) .LOG file. Or can write any custom disk based text file for you in almost any format.

As examples, Distiller easily can analyze web site log files or extract special content from private search engine internals. Or can convert file and font formats. Or can filter all the Fourier transforms from millions of magic sinewaves, keeping only those scant few good ones. Or can design digital filters or solve higher order linear equations.

For final results, you could view your screen displays using Acrobat Exchange, Reader or GhostScript. And use either Exchange or Reader to print. Doing so to both PostScript and non-PostScript printers.

Let's look at one simple example that shows how fast and easy it is to use PostScript as language. Let's find the sine of 35.4 degrees. First, create this textfile by using your favorite editor or WP...

```
%!
% Find sine of 35.4 degrees
35.4 sin ==
```

Drag and drop this file into your Acrobat Distiller. The magic value of 0.579281 should pop right up on your log file display.

Distiller obeys the rules that any *printing* (*print* or *"=="*) commands go to the .LOG file; *writing* commands (such as *write* or *writestring*) output to your custom .TXT disk output file; and any *marking* commands (such as *show*, *stroke*, or *showpage*)

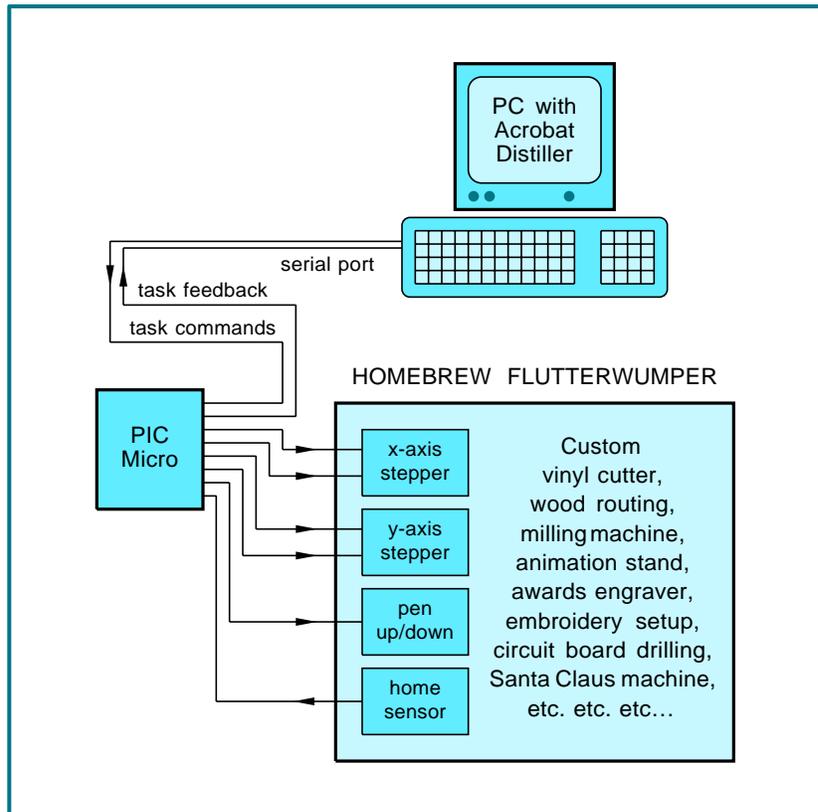


Figure 1 – How flutfile partitioning brings genuine PostScript to PIC flutterwumpers.

go to the generated .PDF file.

Do not end your file with *quit* or control-[D] when using Distiller.

While the log file is intended for error and status messages, you could write nearly anything you want to it at any time for any reason. Sadly, the log file is *not quite* transparent and has a few hidden restrictions. So you will have to decide whether to pick the quick and dirty "free" log file or write a custom output file.

Some log file limitations: Adobe Distiller does a *flushfile* after every newline. Which nicely lets you see problems as they occur. But sure can slow you down. While causing a lot of disk clatter on extended outputs. Long strings are also truncated to 200 chars in the log file. Carriage returns are tricky. And you will get a mix of status, error, and data messages.

One really big time gotcha when writing *any* PostScript as language code that needs any PC disk access: *Always use "\\\" whenever you mean "\" inside of a PostScript string!* The confusion comes about because that "\" acts as a directory boundary in Windows file names, but is a fully reserved PostScript string character.

GhostScript gets around this by substituting "/" for "\" in filenames.

Do not be alarmed when you see a *"Warning: No PDF file produced"* error. If your primary goal is writing your own custom disk file, the oft wanted .PDF viewable file may not be needed at all. Forget the baby; you are after the wash water here.

Note that any Gerber or HPGL to flutfile format disk conversions are a trivial task for PostScript.

```
"0" = Go one step due east
"1" = Go one step northeast
"2" = Go one step due north
"3" = Go one step northwest
"4" = Go one step due west
"5" = Go one step southwest
"6" = Go one step due south
"7" = Go one step southeast

"B" = Begin sequence
"D" = Pen down or cutter on
"H" = Reset to home position
"Q" = Quit sequence
"Rn" = Repeat last n-32 times
"U" = Pen up or cutter off
"X" = Breaking debugger

"% = Ignore comments till cr
```

Figure 2 – One possible flutfile command set. Single characters do elemental tasks.





## SOME FLUTFILE EXAMPLES

Your new PostScript routines can generate flutfiles. Your flutfiles then get used by your flutterwumper.

All in real time, or in a deferred production mode where the same file gets resent once per piece. Again, the big advantage of such a partitioning is that you'll get by with extremely simple and cheap dedicated hardware on the flutterwumper itself.

PostScript is also especially adept in doing the oddball transformations you will need for nontraditional axis flutterwumpers. Such as rectangular to polar or az-el.

Any old terminal program can do your actual comm for you. Plain old *Hyperterm* works just fine.

Let's take a look at three simple PostScript flutfile examples to scope out what they might look like. Since flutfiles are all resolution dependent, let's assume that our flutterwumper resolution is 100 steps per inch.

Or ten mils per step.

In figure three, we make a simple rectangle. First we home with pen or cutter up. Then we shift to the lower left corner of the rectangle and drop the pen. We then trace out the entire rectangular path, raise the pen, and return home.

To get from PostScript into your flutfile, you'll need a PS routine that intercepts PS commands, finds their paths, converts the paths to straight line vectors, resolves these vectors down into machine dependent steps, and then creates a flut file.

All trivial PostScript tasks.

The usual way a PS routine does its flutfile building is by intercepting PostScript's *moveto*, *lineto*, *curveto* and *closepath* operators. Since nearly anything PostScript can be redefined at any time for any purpose, all these operators are redirected to generate the output flutfile commands.

Adobe Acrobat automatically can reduce complex PostScript code into these fundamental *moveto*, *curveto*, *lineto* and *closepath* elements. Your usual way to deal with fancy artwork is to send it on through the *Acrobat Distiller* first. Then print to disk to extract those "just the facts ma'am"

elemental PostScript commands.

Those commands could then get pasted into a PS flutfile generator. Or else get read and interpreted directly from a disk textfile.

Flutfiles can be sent in real time or else saved to disk and reused for production. You can also host create elemental flutfile libraries.

Several ready-to-use examples of suitable tools are in **FLUTOOLS.PS** on the *Flutterwumper Library Shelf* of <http://www.tinaja.com>.

Note that *any* existing PostScript code application can generate all of your desired paths. Although I very much prefer to write custom code in raw PostScript, use of *Illustrator* or any PS-compatible drawing program will work just fine. In fact, *nearly any* existing PS or Acrobat program or code can be used to automatically generate robotic flutfiles.

Thus, *your flutterwumper will automatically speak PostScript from day one*. Royalty free.

Using nothing but a PIC or two.

In this example, our PostScript code is simply...

```
/inch 72 def  
0.4 inch 0.6 inch moveto  
0 inch 0.8 inch rlineto  
1.2 inch 0 inch rlineto  
0 inch -0.8 inch rlineto  
closepath  
stroke
```

A rectangle version using repeats and removal of unneeded characters is also shown. Which dramatically shortens this particular flutfile. Any positioning commands now do take slightly longer to execute since they are limited to 0-7 cardinal moves.

If you are always using inches, you can simply do a *72 dup scale* at the start of your file instead.

Insider secrets of vector-to-step conversion are in **HACK83.PDF**.

In figure four, we create a three inch circle. The PostScript code is even simpler...

```
/inch 72 def  
2 inch 2.5 inch 1.5 inch  
0 360 arc  
closepath  
stroke
```

new from  
**DON LANCASTER**

### ACTIVE FILTER COOKBOOK

The sixteenth (!) printing of Don's bible on analog op-amp lowpass, bandpass, and highpass active filters. De-mystified instant designs. \$28.50

### CMOS AND TTL COOKBOOKS

Millions of copies in print worldwide. THE two books for digital integrated circuit fundamentals. About as hands-on as you can get. \$28.50 each.

### INCREDIBLE SECRET MONEY MACHINE II

Updated 2nd edition of Don's classic on setting up your own technical or craft venture. \$18.50

### LANCASTER CLASSICS LIBRARY

Don's best early stuff at a bargain price. Includes the CMOS Cookbook, The TTL Cookbook, Active Filter Cookbook, PostScript video, Case Against Patents, Incredible Secret Money Machine II, and Hardware Hacker II reprints. \$119.50

### LOTS OF OTHER GOODIES

Tech Musings V or VI . . . . .	\$24.50
Ask the Guru I or II or III . . . . .	\$24.50
Hardware Hacker II, III or IV . . . . .	\$24.50
Micro Cookbook I . . . . .	\$19.50
PostScript Beginner Stuff . . . . .	\$29.50
PostScript Show and Tell . . . . .	\$29.50
Intro to PostScript Video . . . . .	\$29.50
PostScript Reference II . . . . .	\$34.50
PostScript Tutorial/Cookbook . . . . .	\$22.50
PostScript by Example . . . . .	\$32.50
Understanding PS Programming . . . . .	\$29.50
PostScript: A Visual Approach . . . . .	\$22.50
PostScript Program Design . . . . .	\$24.50
Thinking in PostScript . . . . .	\$22.50
LaserWriter Reference . . . . .	\$19.50
Type 1 Font Format . . . . .	\$16.50
Acrobat Reference . . . . .	\$24.50
Whole works (all PostScript) . . . . .	\$380.00
Technical Insider Secrets . . . . .	FREE

### POSTSCRIPT SECRETS

A Book/Disk combination crammed full of free fonts, insider resources, utilities, publications, workarounds, fontgrabbing, more. For most any PostScript printer. Mac or PC format. \$29.50

### BOOK-ON-DEMAND PUB KIT

Ongoing details on Book-on-demand publishing, a new method of producing books only when and as ordered. Reprints, sources, samples. \$39.50

### THE CASE AGAINST PATENTS

For most individuals, patents are virtually certain to result in a net loss of sanity, energy, time, and money. This reprint set shows you Don's tested and proven real-world alternatives. 28.50

### BLATANT OPPORTUNIST I

The reprints from all Don's Midnight Engineering columns. Includes a broad range of real world, proven coverage on small scale technical startup ventures. Stuff you can use right now. \$24.50

### RESOURCE BIN I

A complete collection of all Don's Nuts & Volts columns to date, including a new index and his master names and numbers list. \$24.50

### FREE SAMPLES

Check Don's Guru's Lair at <http://www.tinaja.com> for interactive catalogs and online samples of Don's unique products. Searchable reprints and reference resources, too. Tech help, hot links to cool sites, consultants. email: [don@tinaja.com](mailto:don@tinaja.com)

FREE US VOICE HELPLINE VISA/MC

**SYNERGETICS**  
Box 809-CC  
T Thatcher, AZ 85552  
(520) 428-4073

FREE Catalog: <http://www.tinaja.com>

Finally, in figure five we generate a fancy Palatino "R". Again using any old PostScript artwork creator or else these raw PostScript commands...

```
/Palatino-Bold findfont  
100 scalefont setfont  
100 200 moveto  
(R) show
```

One minor gotcha: The font you pick *must* already be ATM resident in your host. If not, *Courier* will get substituted for misspellings or for a non-Adobe font, while a rectangular bounding box will be substituted for any real but unavailable Adobe font. More details in [DISTPALO.HTML](#) on that *PostScript Library Shelf* of my [www.tinaja.com](http://www.tinaja.com)

A second minor point: *Notepad* has problems with newline control characters, so use *Wordpad* or some other WP or editor instead.

Do note that your flutterwumper electronics could not care less about all this fancy stuff. All your custom hardware has to do is convert ASCII characters into simple steps.

Something that any old PIC can do in its sleep.

## IMPROVING RESOLUTION

Flutterwumper resolution tends to be much poorer than what you'd see in any laser printer. There is not much point in routing out any large wooden display sign closer than, say, a sixteenth of an inch. And it is real tricky to force better than ten mils resolution from any larger homebrew positioning hardware.

A flutfile can be set to most any resolution, so there is no real limit to how fine you slice things. But on any coarser systems, there are a few tricks you can pull that will greatly improve your results.

The first is to *retouch* any grungy flutfile elements so they look best at your chosen resolution. For instance, our initial Palatino "R" looks a little jaggy and unbalanced. The retouched version also shown seems somewhat "less Palatinoish" but more suited to any lower resolution flutterwumper. With some practice, you can easily "sight read" any flutfile. And make these adjustments on the fly.

"Retouching" is slightly different than the usual font hinting processes for coarse bitmaps. Because diagonal paths are permitted. But *Black Book Hinting* techniques might lead to a semi-automated solution.

Retouching is especially useful for character or libraries of forms or other locations where standardized code is going to see lots of reuse.

A second resolution enhancing trick is *stepper phase interleaving*. Most steppers employ four *phases* or increments to complete a step. Each phase moves you one quarter of the way towards your goal.

So, as figure six shows us, instead of jumping a whole X step and then a whole Y step, on any 1-3-5-7 slant moves you first do a quarter X step followed by a quarter y step. All your paths end up much smoother and the jaggies are much smaller.

For low cost larger systems, it is possible to convert any automotive alternator into a cheap power stepper motor. Alternator conversions tend to use three phase steps rather than four. But the same concept of stepper phase interleaving remains.

## FOR MORE INFORMATION

A PostScript-as-language tutorial appears as [DISTLANG.HTML](#) on my *Acrobat Library Shelf*. Find this one at [www.tinaja.com/acrob01.html](http://www.tinaja.com/acrob01.html). Some flutfile generators and utilities similarly should appear over on the *Flutterwumper Library Shelf* at my [www.tinaja.com/flut01.html](http://www.tinaja.com/flut01.html). Lots of fully working PostScript-as-language examples are included. Also offered are PIC sourcecode examples, custom consulting services, and various new design *Infopacks*.

Hot links to both [Acrobat](#) and [GhostScript](#) are also found here. As are extensive PIC tutorials and links.

More details on converting car alternators to power steppers in my [HACK47.PDF](#) A superb alternator conversion video is available from *John Rees*. His address and other mentioned resources appear in the adjacent sidebar.

Your normal starting points for learning PostScript as language are that "red" book, otherwise known as

the *PostScript Reference Manual* and the "blue" book, called the *PostScript Tutorial and Cookbook*. I now stock these and also have the *Whole Works* package of everything important by all major PostScript Authors. Details per my nearby [Synergetics](#) ad.

Contest: A free [Incredible Secret Money Machine II](#) for your best new aps for PIC's and PostScript. 

*Microcomputer pioneer and guru Don Lancaster is now the author of 35 books and countless articles. Don maintains his US technical helpline you'll find at (520) 428-4073, besides offering all his own books, reprints and consulting services.*

*Don has a free new catalog full of his latest insider secrets waiting for you. Your best calling times are 8-5 MST on weekdays.*

*Don is also the webmaster of his Guru's Lair at [www.tinaja.com](http://www.tinaja.com) You could also reach Don at Synergetics, Box 809, Thatcher, AZ 85552. Or you can email [don@tinaja.com](mailto:don@tinaja.com)*

## SOURCES

Adobe Acrobat  
1585 Charleston Rd  
Mountain View CA 94039  
(800) 833-6687

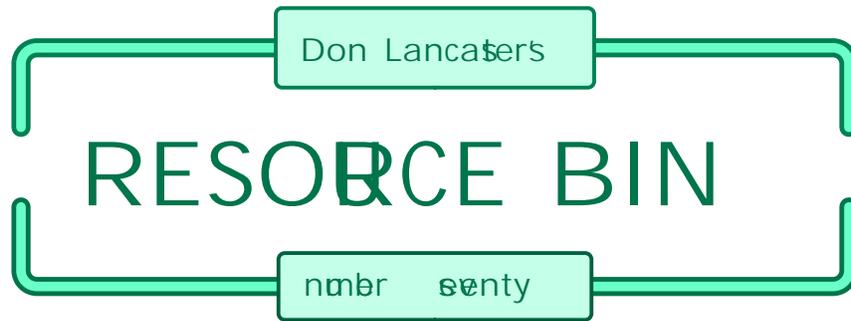
Download Mechanic  
3655 Nobel Drive  
La Jolla CA 92037  
(619) 587-4668

Ghostscript/Aladdin  
Box 60264  
Palo Alto CA 94306  
(415) 322-0103

John Rees  
81 Rees Road  
Santee GA 30571  
(706) 865-5495

Synergetics  
Box 809  
Thatcher AZ 85552  
(520) 428-4073

Videonics  
1370 Dell Avenue  
Campbell CA 95008  
(408) 866-8300



## Exploring PostScript PC Interpreter

Our usual reminder here that the Resource Bin is now a two-way column. You can get tech help, consultant referrals and off-the-wall networking on nearly any electronic, tinaja questing, personal publishing, money machine, or computer topic by calling me at (520) 428-4073 weekdays 8-5 Mountain Standard Time.

I'm now in the process of setting up my new Guru's Lair web site you will find at (where else?) [www.tinaja.com](http://www.tinaja.com) This is the place you go for instant tech answers. Among the many files in our library, you will find complete reprint sets for all of the [Resource Bin](#) and other columns. Plus a brand new [Research InfoPack Service](#).

You will get the best results if you have both [Netscape Communicator](#) and [Acrobat Reader 3.0](#) installed.

### PostScript PIC Flutterwumpers

For quite a long time now, I've been really enthused by Adobe's PostScript computer language. I'll routinely use PostScript well beyond its printing and graphics capabilities. What I find really exciting is that with some brand new happenings, PostScript can now expand into robotics.

Specifically by getting low end PIC microprocessors to speak PostScript!

PostScript and PIC's working with each other might be a killer combo for ultra low cost robotic apps. Especially when such things as reusing existing artwork, interfacing popular design programs, selecting fancy fonts, doing micro-sizing, tool path corrections, or exotic coordinate transformations.

And most of it is simple stuff that you can easily explore all by yourself. Opening up some mind-blowing new opportunities for you.

Others have been quietly extending and greatly improving on PostScript all along. PostScript now offers total web friendliness, transparent video titling apps, document scan conversion,

catalog indexing, interactive forms, internal font provisions, drivers that print anywhere, disability aides, and great heaping bunches more.

But the real biggie is that PostScript itself has recently gotten ridiculously easier to use. And lots more fun for you to play with.

### Getting Started With PostScript as Language

The easiest way to start exploring PostScript is by using a host based interpreter. This is simply a program you run on your PC. You send the interpreter PostScript language code, and the interpreter carries out those commands for you.

Usually with nothing but a simple drag-and-drop.

There are normally three possible results from your PostScript code:

(A) You could generate a file that can be used to print or image or webify or CD ROMify a graphic picture. Such pictures are fully device independent, superbly high quality and an industry standard mix of actual images, text, line art, and fancy fonts.

(B) You can generate a simple log file that quickly gives you important answers directly on your screen.

(C) You can write an output data file to disk. Such an output file can have virtually any format you want it to. In most any language!

These output files can then be used for anything from taking over control of a computer to running a hot tub.

PostScript can similarly get taught to read and act upon just about any input file format you care to. Details

are found in [SIXCLICK.HTML](#).

There are two very popular host based PostScript interpreters. The first and older is that freebie GhostScript shareware from Aladdin Systems. The second is the Acrobat Distiller that is part of the commercial Adobe Acrobat package. By using a student discount, Distiller is available for under \$55.

Get Version 3.01 or higher.

Do not let the name mislead you. Acrobat Distiller is really a full blown PostScript language interpreter that is easily and quickly applied for totally general purpose computing.

When combined with a companion Acrobat Exchange viewer, you'll get everything important that genuine Display PostScript offers. Especially the ability to have source code and fast updating graphic results side by side on your screen at once.

Distiller also acts as "somewhat" of a compiler. In that it automatically reduces PostScript code into the bare minimum "just the facts ma'am" and fast running compressed .PDF format. For instance, an input PostScript proc may include very elaborate fill justify calculations or math-intensive plots. The output PDF format simply shows where to mark the pages or what final numeric data values to output.

Which PostScript interpreter is for you? My own preference is to always use the Acrobat Distiller. Distiller is faster, way easier to run, much better supported, is fully up to date, has only official bugs, and gives you more attractive final results. On the other hand, GhostScript is free and gives you better error reporting.

### Two Examples

Let's look at two examples to get you started off using PostScript as a programming language. Assume you want to find the sine of sixty degrees. Punch this code on into your favorite editor or word processor...

**NEXT MONTH: Don takes another look into secrets of web based research.**

```
%!
% find sine of 60 degrees
60 sin ==
```

...and then save it as either a .PS file or as an ordinary .TXT text file. Drag and drop this file into Distiller or GhostScript, and an output value of 0.866025 should promptly appear in the log window on screen.

We can see several things here. A PostScript file consists of an ordinary text file containing more or less plain English words and numbers. Other more compact file formats are also available for advanced users.

At a minimum, your file must start off with a "%!". To be formally legal, special document structuring convention comments are also strongly suggested for use in fancier projects.

Any line which starts with a "%" is treated as a comment. Thus, only the third line of your program actually does anything useful. This line places a numeric value of 60 onto an internal stack. A sin command then obeys the rule "take the value in degrees on the top of the stack, find its trigonometric sine, and replace the top stack value with this calculated result.

That odd "==" is shorthand for "intelligently print the top stack value to your log file".

Should the PostScript program also produce some graphic or printable output, you simply click on Exchange to view it on screen. Your round trip edit-interpret-view time is typically well under ten seconds and can be done in as little as six mouse clicks.

In PostScript, we have a powerful and quite modern general purpose language that is interpretive (you tell it what to do ahead of time); is stack oriented (everything goes on or off one or more stacks); Polish structured (you give it some values first, followed by your needed instructions that act on those values); is quite object oriented (individual procs can be manipulated with uniformly predictable results); threaded (any command can call any other); weakly typed (most any data structure can get converted to any other); reentrant (commands can call themselves); dictionary intelligent (the grouped key-value pairs play a big role); redefinable (you can add new commands or procs and change any existing ones); and device independent (the same code runs on any host and controls any PostScript device of any quality level).

Let's look at a second and fancier

app. Finding out who accesses your website when is a big deal these days. There's lots of fancy programs (such as WebTrends) which can give you all sorts of exotic plots and graphs. But only with horrendous use of memory and disk space.

PostScript can easily read and then interpret web log files. A no-cost 5K PostScript file can trivially extract log file data that the big programs may miss. Such as analyzing the loyalty of repeat visitors. Or maybe tracking the popularity of a brand new file as it moves up through the ranks.

The site analysis code is a tad too long to show you here. But you can grab examples of these programs as [WEBSITAN1.PS](#) or [WEBLOGU2.PS](#) off my [www.tinaja.com/pslib01.html](http://www.tinaja.com/pslib01.html)

There is extensive documentation inside either file. All you have to do is rename your target and destination files for your own use.

There are dozens more PostScript as language examples where these two came from.

Recapping, PostScript is now really fast, fun, and easy to use. You first create a text file, then drag and drop it into Distiller, and then you optionally view it with Exchange. Reading any file format or creating any new file format or generating graphics.

### Robotic Flutterwumpers

What does all this have to do with robots? Let's backtrack a tad. I like to call a flutterwumper any robotic that moves and either chomps or spits.

Printed circuit drills, sign routers, engravers, silk screen cutters, Santa Claus setups, or animation stands are typical examples.

The basics of flutterwumpers are shown in [FLUTWUMP.PDF](#).

Let's say you decide to build a custom 2-1/2 D flutterwumper. One that has independent X and Y motions in small steps, but only allows a gross "tool up" or "tool down" in your Z direction. Let us further assume that fonts play an important role and that you want to access large libraries of existing symbol icons or whatever. As well as use standard CAD, Mathlab or Illustrator design software.

A PIC or two is the obvious choice to handle all your low level custom flutterwumper controller. But how on earth can you get it to do the fancy stuff? While still remaining both low in cost and elegantly simple.

Your secret lies in the interactions between...

new from  
**DON LANCASTER**

#### ACTIVE FILTER COOKBOOK

The sixteenth (!) printing of Don's bible on analog op-amp lowpass, bandpass, and highpass active filters. De-mystified instant designs. \$28.50

#### CMOS AND TTL COOKBOOKS

Millions of copies in print worldwide. THE two books for digital integrated circuit fundamentals. About as hands-on as you can get. \$28.50 each.

#### RESEARCH INFOPACKS

Don's instant cash-and-carry flat rate consulting service. Ask any reasonable technical question for a detailed analysis and complete report. See [www.tinaja.com/info01](http://www.tinaja.com/info01) for specifics. \$75.00

#### INCREDIBLE SECRET MONEY MACHINE II

Updated 2nd edition of Don's classic on setting up your own technical or craft venture. \$18.50

#### LANCASTER CLASSICS LIBRARY

Don's best early stuff at a bargain price. Includes the CMOS Cookbook, The TTL Cookbook, Active Filter Cookbook, PostScript video, Case Against Patents, Incredible Secret Money Machine II, and Hardware Hacker II reprints. \$119.50

#### LOTS OF OTHER GOODIES

Tech Musings V or VI .....	\$24.50
Ask the Guru I or II or III .....	\$24.50
Hardware Hacker II, III or IV .....	\$24.50
Micro Cookbook I .....	\$19.50
PostScript Beginner Stuff .....	\$29.50
PostScript Show and Tell .....	\$29.50
Intro to PostScript Video .....	\$29.50
PostScript Reference II .....	\$34.50
PostScript Tutorial/Cookbook .....	\$22.50
PostScript by Example .....	\$32.50
Understanding PS Programming .....	\$29.50
PostScript: A Visual Approach .....	\$22.50
PostScript Program Design .....	\$24.50
Thinking in PostScript .....	\$22.50
LaserWriter Reference .....	\$19.50
Type 1 Font Format .....	\$16.50
Acrobat Reference .....	\$24.50
Whole works (all PostScript) .....	\$380.00
Technical Insider Secrets .....	FREE

#### BOOK-ON-DEMAND PUB KIT

Ongoing details on Book-on-demand publishing, a new method of producing books only when and as ordered. Reprints, sources, samples. \$39.50

#### THE CASE AGAINST PATENTS

For most individuals, patents are virtually certain to result in a net loss of sanity, energy, time, and money. This reprint set shows you Don's tested and proven real-world alternatives. 28.50

#### BLATANT OPPORTUNIST I

The reprints from all Don's Midnight Engineering columns. Includes a broad range of real world, proven coverage on small scale technical startup ventures. Stuff you can use right now. \$24.50

#### RESOURCE BIN I

A complete collection of all Don's Nuts & Volts columns to date, including a new index and his master names and numbers list. \$24.50

#### FREE SAMPLES

Check Don's Guru's Lair at <http://www.tinaja.com> for interactive catalogs and online samples of Don's unique products. Searchable reprints and reference resources, too. Tech help, hot links to cool sites, consultants. email: [don@tinaja.com](mailto:don@tinaja.com)

FREE US VOICE HELPLINE VISA/MC

**SYNERGETICS**  
Box 809-NV  
Thatcher, AZ 85552  
(520) 428-4073

FREE Catalog: <http://www.tinaja.com>

## POSTSCRIPT PIC FLUTTERWUMPER RESOURCES

**Adobe PostScript**  
1585 Charleston Rd  
Mountain View CA 94039  
(800) 833-6687

**Scott Edwards**  
964 Cactus Wren Lane  
Sierra Vista AZ 85635  
(520) 459-4802

**Hewlett-Packard**  
PO Box 10301  
Palo Alto CA 94303  
(415) 857-1501

**Parallax**  
3805 Atherton Rd #102  
Rocklin CA 95765  
(916) 624-8333

**Videonics**  
1370 Dell Avenue  
Campbell CA 95008  
(408) 866-8300

**Aladdin/Ghostscript**  
Box 60264  
Palo Alto CA 94306  
(415) 322-0103

**Gerber Scientific**  
83 Gerber Rd  
S Windsor CT 06074  
(203) 644-1551

**Microchip Technology**  
2355 W Chandler Blvd  
Chandler AZ 85224  
(602) 786-7200

**Synergetics**  
Box 809  
Thatcher AZ 85552  
(520) 428-4073

**WEB Trends**  
621 SW Morrison #1025  
Portland OR 97205  
(502) 294-7025

### Flutfiles and PIC's

For some strange reason, I seem to be having some difficulties burning a full PostScript version 2019 interpreter into the 12C508 baby PIC. Mostly, if I did so, it just wouldn't be very fair to Motorola or Intel.

Let's instead partition the problem. Let host PostScript software do the sort of things it is really good at. Let a small and low cost PIC do all your flutterwumper stepper controlling. And let's dream up some way that PostScript can talk to PIC's. And that PIC's can talk back to PostScript.

We have two obvious choices here that certainly should work. Hewlett Packard has long had its HPGL plotter language. And there's the Gerber File Format used on fancier photoplotters. Either of these will work just fine. But they might end up gross overkill for a low cost flutterwumper. And either will tie up a substantial part of your PIC's resources.

Instead, let's get elegantly simple. Let's create a new ultra simple meta language. One that's PIC optimized on one end and PostScript optimized on the other.

I'll call this one a flutfile.

We will limit our new language to ordinary printing ASCII characters and carriage returns in a plain old textfile. We will use the rule of one character per action. Now, this might seem slow and cumbersome, but you usually have more than enough time available on any mechanical motion flutterwumper. And the economics are stunning.

On a 2-1/2 D flutterwumper, there will only be eight possible horizontal actions. You can go one step positive or negative in directions of X, Y, or both at the same time.

So, let the ASCII numeral 0 be east. 1 be northeast; 2 be north. And so on around to a 7 at southeast. To round out your commands, add a U for up, a D for down, and a H for home. And

maybe a few initializing, debug, and repeat commands.

All your PIC has to do is receive a dozen or so serial ASCII characters and then act upon them. For instance, command "0" causes one full step in the positive X direction. Command "1" moves both the X and Y steppers one full step. For max smoothness, your individual partial steps between X and Y can get interleaved. Command "2" moves Y only, and so on.

Now, all that PostScript has to do is accept programming commands and generate a meta file. Which is a lot easier than it sounds. I've got lots of ready-to-run code for you.

Let me briefly outline the PS to flutfile process: The Distiller normally reduces all wanted motions to moveto positioning, lineto straight line paths, curveto curved path generation, and closepath operators. We then eliminate any curveto's with use of PostScript's flattenpath command.

Since any command in PostScript can be redefined at any time for any reason, your remaining stock moveto and lineto commands are intercepted and replaced with your custom code. Among other tasks, this custom code does vector-to-step conversion at the exact resolution needed by your PIC and saves it in a flutfile format.

Detailed flutfile conversion tutorial examples are found in [POSTFLUT.PDF](#), [FLUTOOLS.PS](#), [FLUTDEMO.PDF](#), and [VECTSTEP.PDF](#) on my website.

Included are specific programming details on how you handle boxes, circles, and fancy font characters. You can use these as preliminary test files for your own PIC design.

### How it Works

Here's a summary of how to bring PostScript to any PIC robotic: Install Acrobat Distiller and Exchange 3.01 or better on your host PC. Find suitable project material, from CAD programs that provide a PostScript output, from Illustrator, or by fun writing your own

raw PostScript like I always do.

Use Distiller and a custom software module to convert your pattern files into flutfiles, the simple step-by-step commands that a low end PIC can understand and easily deal with. Such magic tricks as tool path adjusting or fancy coordinate transforms could also be included in this step.

Save your flutfile as an ordinary textfile. Gather up this flutfile with a comm program or whatever and send it to your PIC. Obviously, the flutfiles can be saved for reuse, broken up into module libraries, get web distributed, or combined into projects.

Suitable handshaking takes care of machine busy times.

This two-step process thus brings PostScript to your PIC's simply and cheaply and elegantly.

### For More Help

That Adobe "red" book (PostScript Reference Manual) and their Adobe "blue" book (the PostScript Tutorial & Cookbook) are your standard starting points to learn and love PostScript. These could get combined with my PostScript Beginner Stuff coursework or the Whole Works package that lets you get into PostScript all at once and really big time. Check out my nearby [Synergetics](#) ad for details.

Lots more info on exactly how to bring PostScript to PIC's appear in the previously noted files on my website. For bunches more background, also check the PostScript, Acrobat, PIC, and Flutterwumper library shelves found at [www.tinaja.com](#)

More on the PIC's themselves from manufacturer Microchip Technology, from Parallax, and from Scott Edwards Electronics. Some 600+ annotated hot links to useful PIC sites are found at [www.tinaja.com/pic500.html](#) and also in [www.tinaja.com/picwb01.html](#)

More on those briefly mentioned PostScript transparent video apps by way of [www.videonics.com](#). And, of course, lots more on PostScript and

Acrobat at their [www.adobe.com](http://www.adobe.com).

More technical help, [InfoPacks](#), full consulting, custom programming, or development is available by emailing me through [don@tinaja.com](mailto:don@tinaja.com).

### This Month's Contest

For our contest this month, just tell me what you would do with your PostScript speaking PIC. Especially one of the new minidip baby PIC's.

There should be a largish pile of my new [Incredible Secret Money Machine II](#)

books going to the dozen or so better entries, plus an all-expense-paid (FOB Thatcher, AZ) [tinaja quest](#) for two that will go to the very best of all.

Send all your written entries to me here at [Synergetics](#), rather than to Nuts & Volts editorial. ♦

---

Microcomputer pioneer and guru Don Lancaster is the author of 33 books and countless tech articles. Don maintains his no-charge US tech helpline found at (520) 428-4073, besides offering all of his own

books, reprints, and consulting services. Don also offers a free catalog full of his unique products and resource secrets. The best calling times are 8-5 on weekdays, Mountain Standard Time.

Don is the webmaster of his Guru's Lair at <http://www.tinaja.com>

Full reprints and preprints of all Don's columns and ongoing tech support appear here. You can reach Don at Synergetics, Box 809, Thatcher, AZ 85552. Or send any messages to his US Internet address of [don@tinaja.com](mailto:don@tinaja.com)