SuperPrint's driver that you are using it's spooler, you create printer metafiles (descriptors) that are sent to your temporary directory as specified in AUTOEXEC.BAT. Using a RAM disk for temporary files? ZOOM!

SuperQueue can be running before you send out the temporary files to print, or started later, and used to print files one at a time, in batch mode on command, or automatically as the files are created. Here's the one disadvantage relative to Print Manager: Microsoft's spooler can be started automatically, do its work, and then shut down the same way it started up; you must either start SuperQueue manually as you need it, or through a RUN command in WIN.INI., which consumes memory.

If that were all there was to *SuperPrint*, it would be enough, but there's more. Via its SuperText module, *SuperPrint* provides the ability to create screen fonts for *Windows*, using any of twenty-two included typefaces (really just variations on four - the manual even comes close to admitting that), or the many typefaces of the Hewlett-Packard SoftFonts, Adobe Postscript Type 1, Bitstream, Compugraphics, and URW font libraries. The fonts take up very little disk space, (a set of six sizes ranging in size from ten through twenty-four points typically is about 70 kilobytes small), are generated nearly instantaneously, and because *SuperPrint* translates them to bit mapped graphics as needed for printing, you need no printer fonts.

As replacements for other files, the printer driver and spooler actually **reduce** disk space usage, but as installed, the entire set of utilities occupies about 1 megabyte of disk. We found wry that the documentation explains it's placement of SuperQueue and SuperText's executable programs in the *Windows* directory (you can't alter the choice) by saying that since they were utilities that was the best place for them, yet set up a separate and fairly complicated directory structure for ancillary files, but it's the only negative we can see in the product. We can live with it, and, we imagine, so can you.

SoftType
ZSoft Corporation

If anyone is qualified to give the world an inexpensive source of fonts, it's probably ZSoft. Regardless of what you think of the implementation or even usefulness of their flagship product line -the *PC Paintbrush* family of products - the .PCX graphics file format native to those programs is as close to a standard as there is in graphics. Little wonder, then, that ZSoft has chosen to take their experience and contacts in the area of screen and output layout and provide the *Windows* community with a tool for sprucing up their environment.

SoftType is a screen and printer font generator. Ostensibly, it is designed to work with applications that run under *Windows*, and in fact, there's no way to use the fonts you generate with *SoftType* with DOS-based staples such as *WordPerfect*. Aficionados of Digital Research's *GEM* graphical environment (particularly, users of *Ventura Publisher*), take heart. Even if you don't plan to convert to the new *Windows*-based version of the product, your needs are supported by *SoftType*, also.

Learning to use *SoftType* is almost a non-event. Accompanied by sixty pages of documentation that includes a large amount of reference data, *SoftType* can be mastered in under thirty minutes. The program does almost everything needed to create, catalog, and store fonts automatically. After an installation process that, using its default settings, puts the program files and two of the sixty-plus typefaces provided on your hard disk (total space: about 500 kilobytes), operation of *SoftType* is a matter of choosing the fonts you want to create from a scrolling menu of typefaces and specifying point sizes you want. Before generating your fonts, the program provides you with estimates of the amount of time and disk space that will be needed to complete the task.
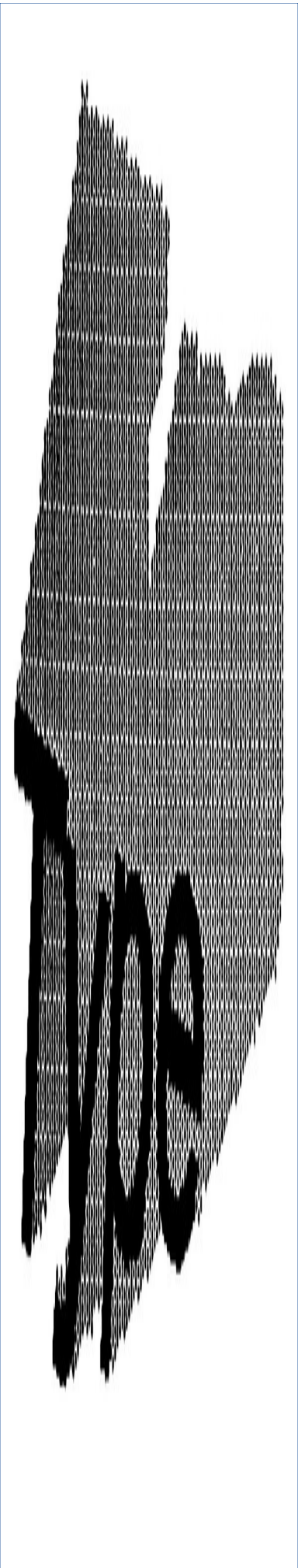
You've just learned as much as you'll need to use *SoftType* effectively. And to think we said it would take thirty minutes! There is, still, one feature of *SoftType* that we haven't covered. You can customize your fonts in a myriad of ways using the STYLE command. You simply set the parameters used by the program, expressed in percentages relative to the standard version of the typeface being modified and degrees of movement, and the only thing limiting what you create is good taste. For example, the illustration shows what happened when we took a straightforward type face and assigned the following values: extension 110%; rotation 9°; slant 6°; gray level 100%; shadow x-axis offset 68%; shadow y-axis offset 42%; shadow size 172%; shadow start and end gray 25%; shadow 3D on; outline width 0%; and outline gray level 23%. All this is a bit confusing, and made more so because not all parameters apply at all times (for example, in the listing above, since there is a zero value for outline width, the outline gray level is immaterial), but a little experimentation will teach you all you need to know.

The general quality of the fonts is very good, but in our tests, serifs displayed a tendency to look slightly spidery. The amount of file space occupied by the fonts is consistent with the *Windows*-resident character sets. If you get tired of the typefaces provided, you also can use those that come with *PC Paintbrush* (.OTL), and .URW files from the German