

Debunking seven common CNC-programming myths.

By the Numbers

► BY MICHAEL DEREN

During the 1970s, NC technology was a major factor in the creation and advancement of modern machine tools. CNCs replaced the old, hard-wired NC systems and are available for virtually every type of machine tool offered.

As machine controls evolved from NC to CNC, certain myths and misconceptions developed as well. The aim of this article is to review—and hopefully dispel—the seven most common myths that have cropped up over the years regarding CNC programmers and their programming tools.

Myth No. 1: Anyone can program a CNC machine. It takes a person with a certain background to program a CNC. First and foremost, a programmer should have worked as a machinist or, at the very least, received an education in manufacturing technology that included some hands-on experience with machine tools in a school's shop or lab.

A novice programmer won't understand the importance and implications of fixtures, feeds and speeds, and depths of cut. Furthermore, it's extremely unlikely that a novice would understand cutting tool or fixture offsets.

The CNC programmer also must be computer-literate. Unfortunately, there are a large number of people at job shops who are computer-illiterate. In addition to comprehending CNC programming software, programmers should un-



As CNC machine use proliferates, so do misconceptions about parts programming and programming tools.

derstand word-processing, spreadsheet and communications software.

And, in this day and age, programmers should know how to navigate Intranets and the Internet, since more and more vendors are disseminating information via their Web sites. Some vendors allow programmers to download drawings of tools and fixture components, which can be added to the programmer's library.

A programmer should possess a solid understanding of data-file management, too. Good file management allows a program and its subcomponents to be found readily.

A tactic that I have used successfully is

to set up a spreadsheet containing every program that I write. I enter all pertinent information onto it, including the customer's name, part number, part name, machine to be used and material type. Then I assign the program a number.

On my company's server is a directory of our machining center programs, turning center programs, etc. (When we purchased a new turn/mill center, I simply added a new directory for that machine.) A subdirectory exists for each of our customers within those directories. The program and its components are stored there.

If I have a repeat job, I do a search in my spreadsheet by part number, material, customer—whatever is appropriate. Then I assign the original program number to the repeat job, and the process continues without a hitch.

Myth No. 2: Programming offline is always faster than programming at the machine. There are times when it is faster to program at the machine than offline. This is often the case with simple workpieces, such as a part produced on a lathe that needs to be faced, turned, bored and cut off, or a square, milled part in which some holes need to be drilled and/or tapped. It's likely that the programmer could do the necessary programming for these parts at the machine in a few minutes.

With an offline CAM system, on the other hand, it may take two to three times as long. The programmer would

have to draw the part or import its rendering, generate the tool paths, post-process the part and send the part information to the shop.

Myth No. 3: All you need to do is program the part, load the program and hit the Cycle Start button. Programmers are not perfect, as much as they would like to believe they are. They do make mistakes. (You can't imagine how hard that is to admit!) Programmers can minimize their mistakes if they are allowed to take the time to verify the program—no matter how simple a part may appear. When the machine operator receives the program, he should dry-run the part to verify the program. After verification, then the Cycle Start button can be hit.

Myth No. 4: All CAM software is pretty much the same. When it comes to CAM software, you generally get what you pay for. CAM software packages range in price from just under \$1,000 to over \$20,000. The lowest-priced software probably doesn't have feature-recognition—the ability to distinguish different sizes of holes, for example—or other important capabilities.

But with costlier software, you might get features you don't need, such as the ability to machine surfaces for mold and die work. Every software package should be evaluated for its potential benefits and ability to meet particular needs. If your shop just turns parts, don't purchase a milling package.

If you mill and turn general parts, software defined as "multipurpose" may suit your shop just fine. That wouldn't be the case if you do mold and die work, though. Multipurpose software would likely lead to large gouges being machined in workpieces, or the software wouldn't be as user-friendly as needed. A mold and die shop will get great results if it programs parts with software designed specifically for its industry.

More importantly, application-specific software is usually easier and

faster to use because, unlike multipurpose software, it incorporates routines geared for certain types of work.

Another factor to consider is whether the software vendor supplies hard-copy documentation. If you have to download documentation from an online PDF file, you might want to consider how long it takes to download and print a 500-page manual.

Myth No. 5: Programmers can learn all they need to know about a CAM software package on their own. If your company lets programmers spend a few days studying the software manuals that come with a new CAM package before they start programming, that's great. But ask yourself this, does self-study make a programmer as efficient as he would be if the software vendor provided a few days of training? The answer is, probably not.

Keep in mind that vendor-provided training not only covers the fundamentals of the CAM package, but also allows a programmer to hear different viewpoints from the instructors and learn program-specific tricks and shortcuts.

If at all possible, the training should be conducted at the vendor's site. Off-site training eliminates the typical distractions encountered at the programmer's shop.

Also, training should be included in the price of the software. What is spent on travel, meals and accommodations will be quickly recouped through the programmer's added proficiency.

Myth No. 6: It's not necessary to spend extra for software updates. Depending on the software developer, the cost of software updates is a worthwhile investment. Granted, a 10 to 15 percent premium for updates can add from \$1,500 to \$2,250 per year for a \$15,000 CAM package. But if the software vendor is making revisions two or three times a year, you'll get your money's worth. Your software supplier is spending a lot of time and money to

develop the additional features and improvements customers are asking for. These improvements translate into reduced cycle times and more efficient programming.

Along with the major revisions, the updates often include fixes for bugs in the previous versions of the software. Some software developers even add trial features in their updates and solicit customer feedback.

The update premium needs to be examined more closely if revisions only occur annually or if the maintenance package just fixes bugs. Check with current users to find out how extensive the revisions are. Keep in mind, though, that if you don't purchase the annual updates/maintenance package, each year you will fall further behind in terms of the efficiencies the software provides, and you will probably lose the technical support the supplier offers.

Myth No. 7: You can't measure the effectiveness of a new CAM package. It's important to know what you're getting before investing in CAM software. Before you purchase any software, take a couple of parts you have already programmed to the vendor and use them as test pieces. Have the vendor program these parts in your presence and compare the programming times. You should see a reduction in time.

If you're buying your first CAM package, the efficiency of programming a part with such software will amaze you. Don't, however, allow yourself to be overwhelmed. Think logically when selecting software, and don't buy into any common misconceptions about CNC programmers or their tools. You'll find little truth in the myths.

About the Author

Michael Deren, a CNC applications engineer who lives in Woolwich, Maine, is a regular contributor and CTE columnist.