

## Tool path plotters that support Custom Macro B

Custom macro B has been around since Fanuc's 6 series controls (6MB and 6TB). It's a very powerful programming tool that has been the topic of many articles here in The Optional stop. (You can find out more about custom macro B on our website, [www.cncci.com](http://www.cncci.com), under resources). Yet for as great a programming tool as custom macro B is, it is still not as widely used as it could be. There are still many people in manufacturing that don't know what it is – or what it can do for them.

For this reason, most suppliers of tool path plotting software have been reluctant to support custom macro B. Again, if only a small percentage of their customers use custom macro B, they're not going to place too much emphasis on developing support for it.

Unfortunately, custom macro B programs can be quite challenging to verify. And if program verification must be done at the machine (without a PC based tool path plotter), a great deal of machine time can be taken (wasted) while verifying custom macro programs. Many programmers try to verify custom macros when the machine is down (during breaks, lunch, and off-shift periods) to minimize the impact on production, but this can be cumbersome. Some controls have tool path plotters built in, which really helps, but still verification of custom macros must be done at the machine.

For years, we've been searching for a tool path plotter that can support custom macro B programs. As with the verification of normal CNC programs, this would allow programmers to verify custom macro B programs off line.

On our website and in our CNC Tech Talk forum (again from [www.cncci.com](http://www.cncci.com)), we've just had a series of posts from people who have located tool path plotters that support custom macro B programs. We list them here. Note that we have had no first hand experience with any of them – so we're not comparing or reviewing them in any way. Nor are we supplying pricing. Again, we're just listing them. If you know of others, we'd like to hear from you. We'll publish them in a future issue of The Optional Stop.

- Vericut by CG Tech ([www.cgtech.com](http://www.cgtech.com))
- Flexlink 2000 by Software Magic ([www.softwremagic.com/products&services/flexcam/flexcam\\_link.htm](http://www.softwremagic.com/products&services/flexcam/flexcam_link.htm))
- Predator CNC editor by Predator Software ([www.predator-software.com](http://www.predator-software.com))

One person posting to our CNC Tech Talk forum says he verifies the arithmetic and logic using the BASIC programming language. Another says he does the same using Microsoft Excel.

M01

## Tapping on a turning center (without canned cycles)

Unless your Fanuc controlled turning center came with live tooling, it's likely that you don't have canned cycles (G80-G89) like those you find on machining centers. You might have G74 and G75, the multiple repetitive cycles for grooving (or cross drilling) and peck drilling, but nothing that will help with tapping.

While this might be considered a severe oversight on Fanuc's part, it is not at all difficult to program a tapping operation for machines that don't have a tapping cycle. If you're machine does-

n't have a C axis (or indexer) in the spindle and live tooling, you'll only be tapping one hole – at the workpiece center – so programming will not be very difficult.

Here are the commands for tapping, starting with the turret index command that selects the tap. Be sure to read the important notes that follow prior to using this program segment.

- M065 T0505 M41 (Select 1/2-13 tap, select low range)
- M070 G97 S500 M03 (Start spindle at 500 RPM)
- M075 G00 X0 Z0.25 M08 (Rapid to approach position, start coolant)

### In This Issue:

<i>Tool path plotters that support Custom Macro B</i> .....	1
<i>Tapping on a turning center (without canned cycles)</i> .....	1
<i>Should you use Parametric Programming?</i> .....	2
<i>Product corner</i> .....	4

### Subscription Information:

The Optional Stop is published quarterly by CNC Concepts, Inc. and is distributed free of charge to people downloading it from our website ([www.cncci.com](http://www.cncci.com)). Back issues are available, but a \$10.00 charge per issue will apply to any back issues not currently published on our site. Back issues 1-50 are available for a total of \$199.00.

All techniques given in this newsletter are intended to make CNC usage more efficient, easier, and/or safer. However, CNC Concepts, Inc. can accept no responsibility for the use or misuse of the techniques given.

```
N080 G01 Z-1.0 F0.0769 (Feed tap into
hole)
N085 M04 (Reverse spindle)
N090 Z0.25 (Feed tap out of hole)
N095 M03 (Reverse spindle back to for-
ward, assuming next tool requires for-
ward direction)
N100 G00 X8.0 Z8.0 (Move to tool
change position)
N105 M01 (Optional stop)
```

There are a few important notes to make about this program sequence.

1) In line N065, we're selecting the low spindle range. Most machine tool builders recommend tapping in the low range (if your machine has more than one spindle range).

2) In N075, note the 0.25 approach position in Z. We're assuming that you are using a floating tap holder that allows the tap to float along the Z axis (this is the same type of tap holder used on a machining center that does not have the rigid tapping function. This kind of holder is required if the turning center cannot precisely synchronize the spindle reversal and with the Z axis motion. We've seen people get away without a floating holder, especially for larger taps.

3) In line N080, notice that the feedrate is set to the thread's pitch. For threads in the English (inch) system, pitch is calculated by dividing one by the number of threads per inch. Since we're using a 1/2-13 tap, our pitch will be 1/13, or F0.0769.

4) In N085, notice that we've placed the spindle reversal command in a line by itself. Turning centers vary dramatically when it comes to how they make their M codes work. With some machines, the M code will take effect at the beginning of a motion command. With others, it will take effect at the end of a motion command. It is possible that you can place the M04 command in line N090 (if the M04 takes effect at the beginning of motion). Our best suggestion here is to cautiously watch the tapping operation (without a part and without dry run turned on) to see that the tapping motion looks okay. If it does not, try

moving the M04 to line N090. Again, don't blindly try this program with a workpiece in position for the very first try!

M01

## Should you use parametric programming (custom macro B)

While monitoring the various manufacturing-related newsgroups and forums, I see a lot of discussions (arguments?) related to the wisdom of using parametric programs (Fanuc calls their version of parametric programming custom macro B). People seem to get pretty passionate when it comes to defending their position.

Determining whether to use parametric programming techniques is an extremely important decision. It should be based upon whether you have a good enough application for parametric programming to justify its use. Remember that there are some negatives to using parametric programming. We mention two of them. First of all, it's not free. Most control manufacturers classify it as an option that your company must pay extra to get. While one or more of your current machines may have parametric programming, you may have to special order it for others – and be sure you order it with future machine purchases.

Second, parametric programming commands tend to be rather cryptic and difficult to understand. It is not uncommon that but one person in a company will learn enough about parametric programming to use it. If this person leaves the company, it can spell trouble for others. If you incorporate parametric programming methods in your programs, be sure that several people understand its use.

For the right application/s parametric programming can provide fantastic benefits that will easily overcome the negatives. Frankly speaking, almost every company has some

pretty good applications for parametric programming – but they may not know it. And unfortunately, you have to learn quite a bit about parametric programming in order to determine whether you have a good enough application to justify its use (which helps explain why it is not widely accepted).

All programmers should learn about parametric programming even if they don't have good applications for it. Truly, the first step to doing anything is knowing what's possible. If you don't understand the applications, features, and functions of parametric programming, how can you make an informed decision as to whether or not it should be used?

(If you feel at all weak in your parametric programming skills, there is a relatively inexpensive way to learn more about it. We've just introduced an on-line course for (\$89.00) that describes parametric programming in detail. It's self-paced and on-going, so you can sign up and begin at any time.)

In very general terms, applications for parametric programming fall into five basic categories. We provide more information about these categories on our website ([www.cncci.com](http://www.cncci.com)). Look under RESOURCES and CNC TIPS, for the article What is Parametric Programming?

- Part families
- User-created canned cycles
- Utilities
- Complex motions
- Interfacing with machine accessories

The classic (most commonly known) parametric programming application is part families. One general purpose program can be used to machine any number of workpieces in the family. Variables (commonly placed at the beginning of the program) are used to specify the values for things that change from one part in the family to another. These values are referenced at any time they are needed in the program. So by changing a variable (say, for a part's overall length), the program will behave dif-

ferently – machining the part as specified.

But for as important a category as part families is, it is not the only application category for parametric programming. Many people mistakenly believe that if they don't have any part families, they don't have an application for parametric programming. Again, this is a mistaken belief.

The user created canned cycle application category is also important. If you find yourself wishing for more canned cycles (like the tapping cycle discussed in an earlier article of this newsletter issue), or if you don't like the way your current canned cycles work (many people don't like the way the G83 deep-hole peck-drilling cycle works because it doesn't allow different peck depths), you can create your own canned cycles. Examples include a grooving canned cycle for turning centers and a thread milling canned cycle on machining centers.

Utility applications include anything that enhances the way the CNC machine behaves. Countless things can be done to minimize setup and cycle time, make the machine easier to work with, and make the machine more fail-safe to run. Say for example, your turning center setup people are constantly having to bore soft jaws. A special utility program can be created to allow them to do so by simply specifying a few variables related to the size of jaws they currently need to bore.

Complex motion applications include providing the ability to create your own interpolation types. You know that machines commonly come with linear and circular interpolation. For thread milling on a machining center, there is also helical interpolation. But say you have to mill a tapered thread. This requires a kind of "spiral interpolation". The tool must make a spiral motion in XY (not a simple circular motion as is required for straight threads) while making a linear motion in Z. Some newer controls allow this, but if you have a control that does not, you can generate the needed

motion with parametric programming.

As the name implies, the interfacing with accessories category allows you to make CNC programs work with accessory devices. Spindle probes on Fanuc controlled machining centers, for example, rely heavily on custom macro B. But remember, if you have an accessory that you need to control/talk with through a CNC program, you can do so with custom macro. Post process gauging systems, for example, can feed data back to the control to cause offset changes based upon the findings of the post process gauge.

We've just scratched the surface of what can be done with parametric programming. If you have never attended formal training related to parametric programming, we urge you to learn more. GE Fanuc offers classes on a regular basis. As do some of the more progressive machine tool suppliers. And again, you can attend our on-line class. Unless you know all that is possible, you can't make an informed decision related to whether or not to use it in your specific situation.

M01

## How does cutter radius compensation work?

Machining center programmers can program contour milling cutters in one of two ways. They can either program the cutter's centerline path or program the work surface path. Either way, the feature cutter radius compensation will provide a great deal of flexibility to the person running the machine. While cutter radius compensation is a basic and important feature – one that is relatively easy to program, many (especially novice) programmers don't understand how it works.

To begin, consider a program for a 1.0 inch diameter end mill that does not use cutter radius compensation. The programmer must program the

milling cutter's centerline path, considering the milling cutter's 0.5 inch radius for every position. If machining the right end of a 6.0 inch long workpiece, for instance, an X position of X6.5 will be used. Here are a few commands for this applications:

```

.
.
.
N065 T05 M06 (1.0 end mill)
N070 G90 G54 S400 M03 (Select abs
mode, coord. system, and start spindle)
N075 G00 X6.5 Y-0.6 (Rapid to approach
position in XY)
N080 G43 H01 Z-0.6 (Rapid to work surface
in Z)
N085 G01 Y4.5 F5.0 (Mill right end)
N090 G00 Z0.1 (Retract in Z)
N095 G91 G28 Z0 M19 (Retract to tool
change position)
N100 M01 (Optional stop)
.
.
.

```

Again, in line N075, we considered the radius of the milling cutter in its approach positioning move, keeping the tool 0.5 away from the X surface and clear of the workpiece in Y.

Now consider a program that machines this same right side and uses cutter radius compensation (programming the work surface path).

```

.
.
.
N065 T05 M06 (1.0 end mill)
N070 G90 G54 S400 M03 (Select abs
mode, coord. system, and start spindle)
N075 G00 X6.8 Y-0.6 (Rapid to prior approach
position in XY)
N080 G43 H01 Z-0.6 (Rapid to work surface
in Z)
N083 G42 D31 X6.0 (Instate cutter radius
compensation)
N085 G01 Y4.5 F5.0 (Mill right end)
N090 G00 Z0.1 (Retract in Z)
N093 G40 (Cancel cutter radius compensation)
N095 G91 G28 Z0 M19 (Retract to tool
change position)
N100 M01 (Optional stop)
.
.
.

```

As you probably know, when approaching we must keep the tool away from the surface to be milled. I call this initial position the prior position

(it is in line N075 of the previous program). We're keeping the tool 0.3 away from the surface to allow for a range of cutter sizes. If the setup person does not have a 1.0 diameter cutter, they can use something slightly larger – up to 1.6 in diameter in this example.

In line N083, cutter radius compensation is being instated. Since this program is specifying work surface path (X6.0 is the work surface), we're depending upon the setup person to enter the cutter's radius (0.5 in our case) in the cutter radius compensation offset. In this example, offset number thirty-one (specified by the D word in line N083) is the offset into which the setup person will enter the value 0.5 (if they are using a 1.0 diameter cutter).

All that is really happening in line N083 is that the control will keep the milling cutter away from (on the right side of) the X6.0 surface by whatever it sees in offset number thirty-one – a value of 0.5 in our case. Note that this totals the 6.5 value used in the program that doesn't use cutter radius compensation.

Once cutter radius compensation is instated, it remains in effect until it is cancelled (by G40). It is best if you can retract the tool in Z prior to canceling (though this may not always be possible).

Note that this is an extremely simple example – just milling the right end of a workpiece. But keep in mind that this same principle applies to the milling of more complex contours. Once instated, the control will continue to keep the milling cutter on the right or left side of all surfaces it sees coming up in the program.

M01

## Product Corner

### New on-line course: Machining center setup and operation

Learn how to setup and run the most popular type of metal-cutting CNC equipment!  
Task-oriented approach  
Six activity-packed lessons  
Printable course text (over 130 pages)  
On-going, self-paced course - enroll and begin at any time!  
Personal assistance from your instructor,  
View/print (requires Adobe Acrobat Reader - download this reader below)  
\$49.00

CNC machining centers are among the most popular metal-cutting machine tools in use today. Almost every company that manufactures any metal products has more than one – and is constantly searching for qualified people to run them. To fully master the use of a CNC machining center, a person must become proficient in three distinct skills – programming, setup, and completing production runs. This class addresses two of the three; setting up a machining center and how to complete a production run (another class is available for helping you learn CNC machining center programming).

The primary goal of this course is to help you learn enough about CNC machining center setup and operation that you can secure an entry level position with a CNC-using company. If you're already working for a CNC using company, the secondary goal is to increase your level of knowledge about CNC machining center usage. If you're currently working as a CNC operator (not setting up machines), this course will show you what it takes to step into a setup person's position.

### Managers!

Progressive managers are always looking for ways to improve the proficiency of their staff. Considering the fact that most entry-level CNC people are hired-in with little or no previous manufacturing (let alone CNC) experience, CNC-using companies are usually on their own train people from scratch. Many have done a pretty good job when it comes to providing training for basic skills that CNC operators require - skills like blueprint reading, tolerance interpretation, gauging, and shop math - to name a few.

When it comes to setup and maintaining production runs for CNC machines, however, companies vary with regard to how well they acquaint their CNC people with the machines. Why not enroll new people in this class to allow them to become familiar with the principles related to setting up and running CNC machining centers?

And by the way the best CNC operators are eventually promoted to become setup people - and the best setup people are commonly promoted to become programmers. Yet you may be having trouble finding resources to help you upgrade the skills of your operators and setup people to programmer level. You may, for example, want your operators to be able to make minor program modifications to handle speed and feed problems. Or you may want setup people to completely verify new programs by themselves. Or you may simply want people in your company (including managers, design engineers, and other engineering people) to know more about CNC programming. Why not enroll them in our on-line programming courses to help them learn the basics of CNC G code level manual programming? Better yet - why not allow them to work on course activities for part of each work day?

Since we grade students after every lesson, you can easily track performance. When a student successfully completes one of these classes, you can rest assured that they truly understand course content.