CNC Trade Secrets

A Guide to CNC Machine Shop Practices

James A. Harvey

CNC Trade Secrets

A Guide to CNC Machine Shop Practices

James A. Harvey

INDUSTRIAL PRESS

A full catalog record for this book is available from the Library of Congress.

ISBN 978-0-8311-3502-7

Industrial Press, Inc.
32 Haviland Street
South Norwalk, Connecticut 06854

Sponsoring Editor: John Carleo
Developmental Editor: Robert Weinstein
Interior Text and Cover Design: Janet Romano-Murray

James A.Harvey
12112 St. Mark St.,
Garden Grove, CA 92845
info@industrialpress.com

Manufactured in the U.S.A.

10 9 8 7 6 5 4 3 2 1

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I'd like to thank my dad, who blessed me with some of his extraordinary mechanical aptitude, and my mom, who put up with both of us constantly tinkering with some contraption. My daughter Joanna, who at a young age helped me transition into the computer world, and my son Billy, who on the darkest of days, always manages to bring a little sunshine.

I'm thankful for the support of Robert Weinstein, whose tenacity, patience, and forward thinking turned this book into a reality. I'd also like to thank Janet Romano and John Carleo from Industrial Press for their kind support.

And a salute to all the technical people, machine builders, and software developers, who provide the remarkable tools we use in the shop and elsewhere that make our lives easier and more productive.

INTRODUCTION

I'm spoiled. With few exceptions, I now put everything I can now on the CNC machines. In most cases, jobs are less labor intensive, less prone to error, and can be completed sooner than when using conventional machines. With the user-friendly CNC technology available today, even making one or two simple parts is often easier and more efficient than conventional machining.

Does that mean the end of conventional machining is near? I don't think so. There are certain tasks you can do with conventional machines that CNC machines are just not good at, such as "blending." CNC machines can't blend flawed features very well simply because they don't have eyes. They run on numbers. Many jobs that come through the door don't have exact numbers to work to or any documentation for that matter.

Often parts that need rework are so beat up, worn out, or outdated that, even if you had a print, there's a good chance the dimensions on the print won't match the part. Maintenance machining and mold repair jobs come to mind as jobs that often need rework and manual blending of some sort.

There are craftsmen in our shop who know little about CNC machining and CNC people who are not craftsman. The funny thing is that often these people imagine the other as having some kind of God-like abilities. The fact of the matter — little would get accomplished it weren't for these two groups working together.

CNC machining involves a combination of three things:

1. Machining knowledge
2. Controller familiarity
3. Programming knowledge

Machining is the art of cutting away material in the proper sequence, selecting and sharpening cutters, setting feeds and speeds, measuring, and determining how you are going to hold the work. The planning and cutting characteristics of conventional and CNC machining are quite similar with the exception that you can machine parts a lot faster using CNC machines because they can read numbers quickly and move accordingly. They also don't take coffee breaks.

Programming is essentially the art of making the machine move the way you want it to.

CNC programming, at its core, is simply point-to-point programming which in turn obliges the cutter to move from point to point. You don't have to be especially adept at doing mental gymnastics to program CNC machines because almost all the instructions you give the machine are sequential. CNC programs are a lot like driving instructions: "Go north three miles then turn left." "Go west ten miles then turn right." Etc.

My previous book *Machine Shop Trade Secrets* deals mostly with conventional machining.

The practice of starting out on conventional machines to learn machining may be shifting now as CNC machines become increasingly more popular, less expensive, and easier to use. If you are an experienced conventional machinist, you'll have the advantage of being able to concentrate your efforts on learning the controller and how to program. I don't believe it is essential to have experience on conventional machines to learn CNC machining. However, a person with a strong conventional machining background will likely have an easier time getting good parts consistently.

To the untrained eye, CNC machines may look complicated. They did to me at first. I was so accustomed to machining everything conventionally,; I couldn't imagine how anybody could machine parts simply by pressing buttons on a control panel.

When I signed up for a CNC programming course years ago, I asked the instructor how long he thought the average person would need to be able to set up and operate a CNC machine. His answer surprised me. He said " a couple of days." In my ignorance, I figured it would take at least a couple months of intense training. As it turns out, he was much closer to being correct than I was.

The other day while running a job on the CNC mill, I took some time to look over the list of G codes posted on the controller. To my surprise, I wasn't familiar with a lot of the codes. I have literally programmed and run thousands of different parts through these machines. "How can that be?" I asked myself. The only logical conclusion I was able to come up was this: You don't have to know "everything there is to know" to be able to make parts on these machines. It is sort of like using Microsoft Word. You don't have to know everything there is to know about Microsoft Word to be able to write a letter.

This book is written from a machinist's perspective. I work as a machinist in a small support shop. What I strive for and demand of myself is being able to accomplish whatever machining job comes through the door. If I don't know some aspect of doing a job such as thread milling, I'll learn it. I'm stubborn in that regard. I may forget what I learned a few weeks later, but that's another story.

I have always preferred learning something new by doing a project. By doing a project you immediately to start to separate what is important from what is not.

The first chapter talks about planning jobs. The planning stage is where you may save countless hours of time and frustration later on. The second chapter is devoted to "work holding" which is often a challenging aspect of machining. The third chapter discusses ways to avoid crashes. Nobody likes to crash. The fourth chapter goes into some detail about applying edge dressings to parts. The fifth chapter discusses something dear to my heart, which is helping designers and engineers help us. The sixth, seventh, and eighth chapters are devoted to familiarizing yourself with CAD/CAM systems and G code. Once you get over the pain of learning a CAD/CAM system, you'll never want to go back. In these chapters we'll be modeling, programming, and machining a simple part. The methods applied to make this part are used over and over in shops and provide a solid base of knowledge from which to expand.

The ninth chapter provides an overview of the commonly used controls on CNC machines. The tenth chapter provides miscellaneous tips that may help you through your day in the shop.

The CNC machines and controllers discussed in this book are primarily Haas. The only reason for that is because Haas is what we use in our shop. The setup and programming procedures discussed are specifically for Hass machines, but will likely be useful in a generic sense for other brands of CNC machines. Nevertheless, if you don't use a Haas right now, chances are you will in the future. At this time, Haas is outselling their competitors three to one.

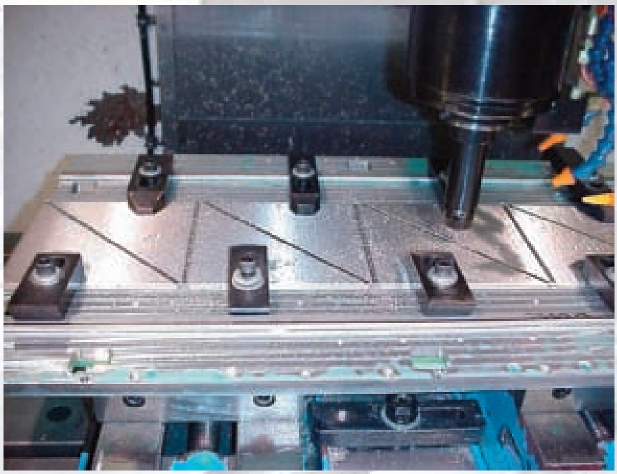
Let's get started.

CHAPTER 1

Proper Planning

When I began machining, CNC (computer numerical control) machines were just coming into mainstream use. The first few shops I worked in had only conventional machines. It wasn't until the mid-1980s that I started to notice the incredible versatility of CNC machines. The contouring and shapes the machines could generate made many jobs much easier. The use of templates to file and sweep in surfaces was rapidly becoming a thing of the past.

Initially, I was intimidated by the technology. I couldn't imagine how a person could machine parts simply by pushing buttons on a control panel. The mysterious code that ran the machines seemed to be the domain of computer people; it was seemingly beyond my comprehension. It wasn't until later, when I took a class in programming, the mysteries started quickly dissolving.



As versatile as CNC machines are, they are only as good as the people programming and operating them. The cliché "garbage in garbage out" is well suited to CNC programming and machining. There is simply no substitute for proper planning and machining know-how. A programmer without much machining experience would likely struggle to produce good parts consistently.

One of the great virtues of CNC machines is that, while they are running, the machinists or operators can be working on other tasks. In essence, CNC machines take the labor out of machining.

Another virtue of CNC machines is that the cutting and measuring process so prevalent in conventional machining is virtually eliminated. If programmed and set up correctly, the cutter will go precisely where it should so that dimensional accuracy comes quickly. Usually only minor adjustments are needed to compensate for slight variations in tool size.

For anybody who wants to learn the machining trade, gaining experience on conventional machines has advantages, but it's not entirely necessary. I was recently involved with teaching an inexperienced industrial engineer how to use our CAD/CAM system and CNC machines. Once he became familiar with the software and the different feeds and speeds that could be used with the various cutters and materials, he became an asset to our shop.

1. Initiate a new project with proper planning. (see Fig. 1-1 and Fig. 1-2)

It is one thing to create a beautiful model or drawing on a computer. It is quite another to figure out how you are going to hold the part for machining.

Planning involves a variety of decisions that have to be made before any material is cut. Table 1-1 lists many of the kinds of decisions that must be made.

Because of the infinite variety of machined parts — especially milled parts — it is somewhat difficult to categorize how to go about planning. There are enough similarities, however, that certain techniques can be used successfully time and again.

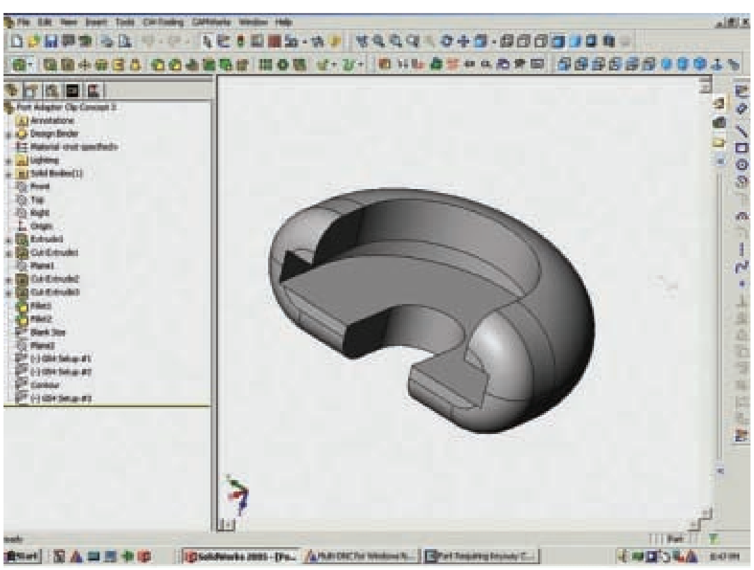


Figure 1-1 Parts are often easier to model than they are to machine.

Table 1-1 Planning Decisions

1. How is the part going to be held? Will a fixture be needed?
2. What sequence of setups will be used?
3. What size raw stock are you going to start with?
4. What size cutters will you choose and are they available?
5. Will squaring be done beforehand or in the program?
6. Is the part rigid or will flimsiness be an issue?
7. What feeds, speeds, and depth of cut will you choose?
8. Where are the G54 starting points going to be placed?
9. Will the part have to be roughed in first before finishing to avoid warping issues?

Good planning is key to having jobs run smoothly. One paradox that machinists face is if a job runs smoothly, nobody pays much attention. Therefore, any time you planned and executed a job that runs smoothly, you should give yourself credit.

As a beginner, there will be no shortage of people willing to offer advice on what they think is the best way to run a job. It's tough being a beginner. There are an infinite number of ways one could run a job; many have no advantage over the other. The bottom line? If a job runs smoothly, is in tolerance, and is completed in a reasonable amount of time, then you have accomplished your mission!

With that being said, it's always humbling when you think you've planned a job well and it turns into a nightmare. I believe planning, programming, choosing speeds, feeds, and depth of cut before cutting anything are basically educated guesses. Yet, the more experience you gain, the better your guesses become.

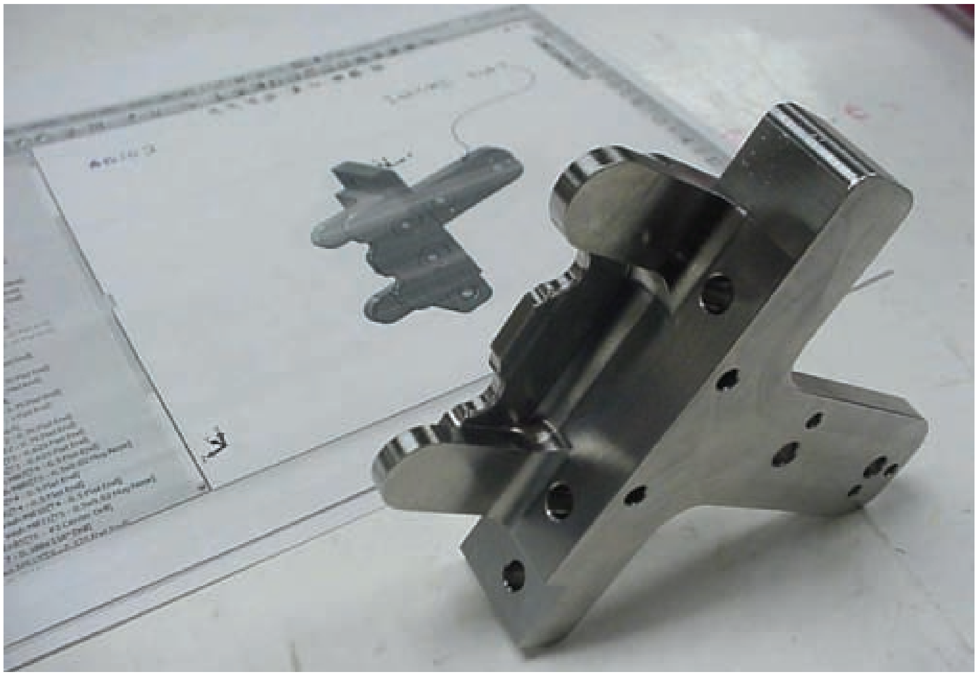


Figure 1-2
Some parts can be planning nightmares.

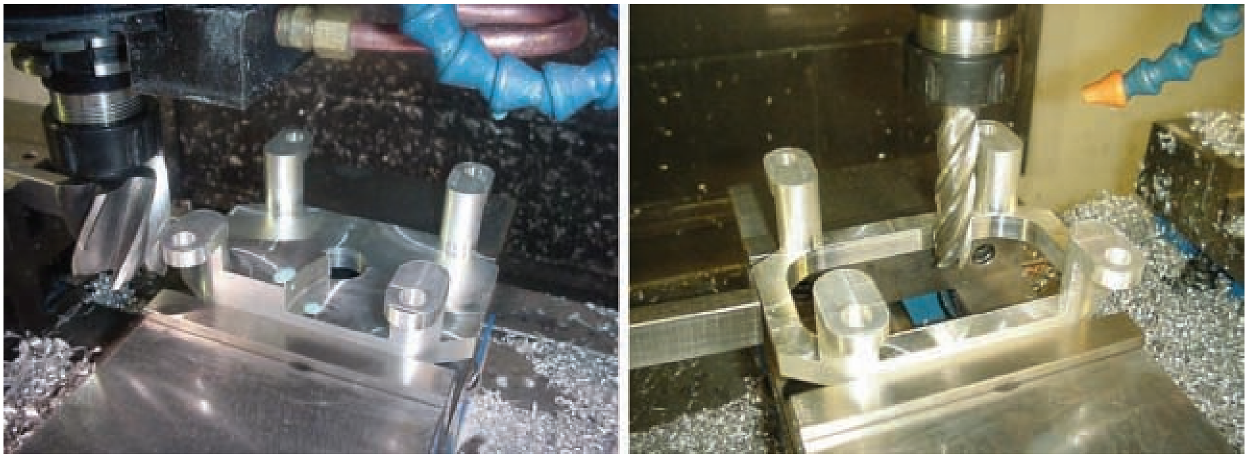


Figure 1-3 This setup was chosen based on using the longest contact area possible with the vise jaws.

2. Use the longest contact area. (see Fig. 1-3)

One of the first and simplest things to look for is the length of the contact area between the part and the vise jaws. In general, choose the longest contact area possible. The setup on the left of Figure 1-3 is rigid before any material is cut away. Initially, aggressive cuts can be made with high feed rates to remove material. As the “window” cut shown on the right nears completion, the part becomes much weaker. Lighter cuts must be used at this stage. These types of machining scenarios are common and must be planned for in advance.

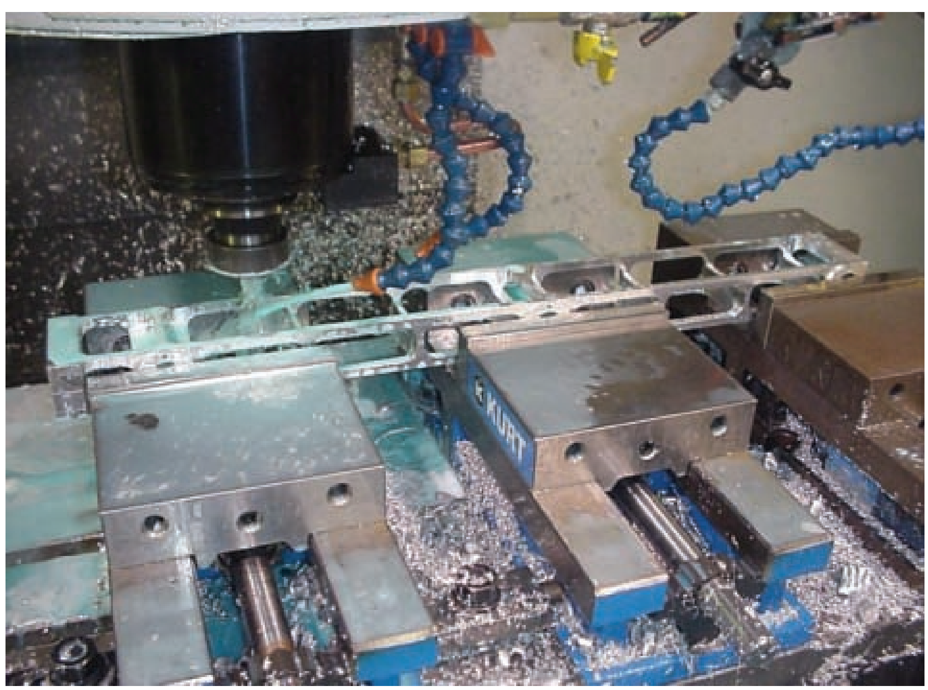


Figure 1-4 The deep windows in this part were cut first because that’s when the part had the most rigidity. The shallower windows being cut here are cut last because the part is significantly weaker.

3. Rigidity is an important factor. (see Fig. 1-4)

As you envision or simulate material being cut away, think about the rigidity of the part and setups. When possible, choose a sequence where you can use short, sturdy end mills.

Long end mills are more sensitive and unforgiving than short end mills to feeds and speeds. They usually need to be run slower. Long end mills have a tendency to chatter and push away. If you need to use long end mills to cut tall or deep features, then sequence the setups so that the long end mills are run when the part has the most rigidity. If you use long end mills on a part that has already been weakened by previous machining, you'll likely compound difficulties.

4. Spend time visualizing. (see Fig. 1-5)

Planning begins by visualizing the different ways you may be able to hold a part for each setup. As you think about the different possible setups and machining operations, keep track of what material remains on a part that can be used to hold the part for future setups and also what material remains so you don't run into any uncut material by mistake. Some parts may not have simple solutions. Look for a sequence that uses the least amount of setups. Multiple setups take time and are ripe for inducing dimensional errors. CAD systems help you visualize all the possible holding combinations because you can easily flip and rotate the model on the computer screen. In general, I prefer doing a job that uses the fewest setups possible, even at the expense of having to run long end mills less aggressively or cutting angles by 3D milling.

Figure 1-5 Try to plan jobs using the fewest setups possible.

5. Determine if you can machine a job holding the material in a standard milling machine vise. (see Fig. 1-6)

CAD systems help you visualize all the possible holding combinations because you can easily flip and rotate the model on the computer screen.

Plan to use a vise when possible. A vise eliminates a lot of variables such as the accuracy and rigidity of your work-holding device. A good quality vise is both accurate and rigid.

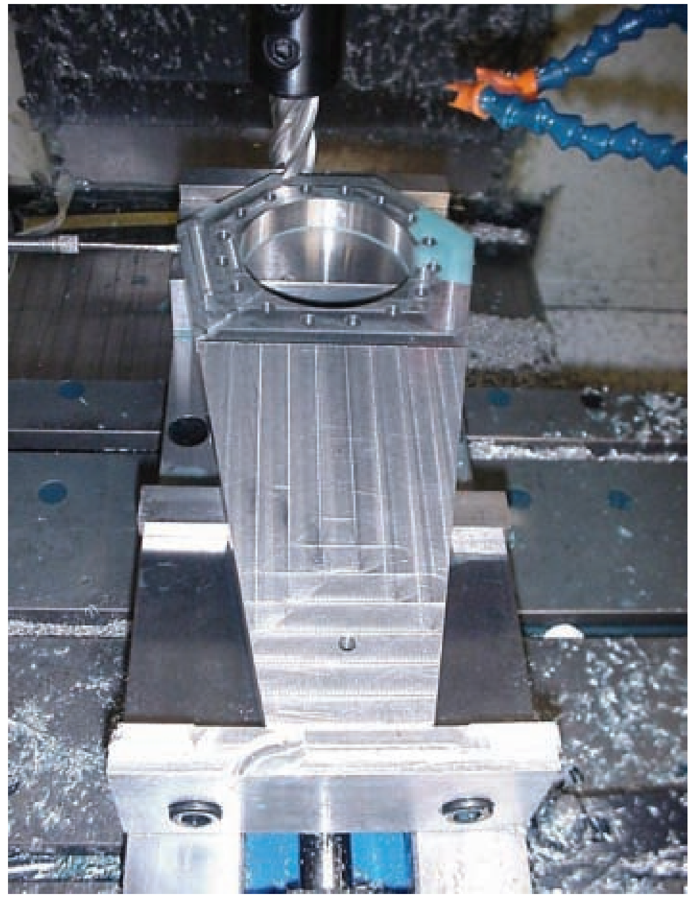


Figure 1-6 Plan to use a vise when possible. Holding parts in a vise eliminates the need for time- and material-consuming fixtures.

6. Face off the backside of a part to finish the overall thickness. (see Fig. 1-7)

Often when using a vise to hold a part, you will at some point have to face off the backside to bring the part to the correct thickness. This is a technique I use often. It eliminates the need for a fixture and allows you to do a lot of machining in one setup. Furthermore, the material thickness you start with is not critical within reason. Generally 1/8″ of material is plenty to hold in a vise.

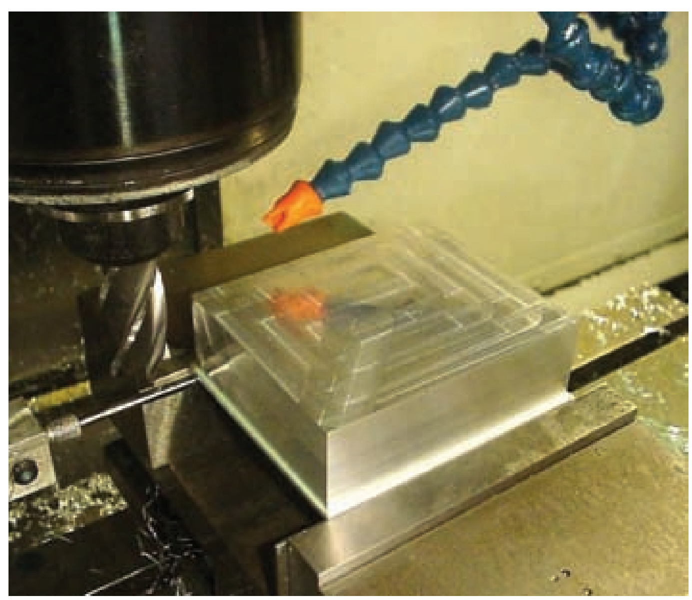


Figure 1-7 1/8″ of material is generally all you need to hold parts securely in a vise.

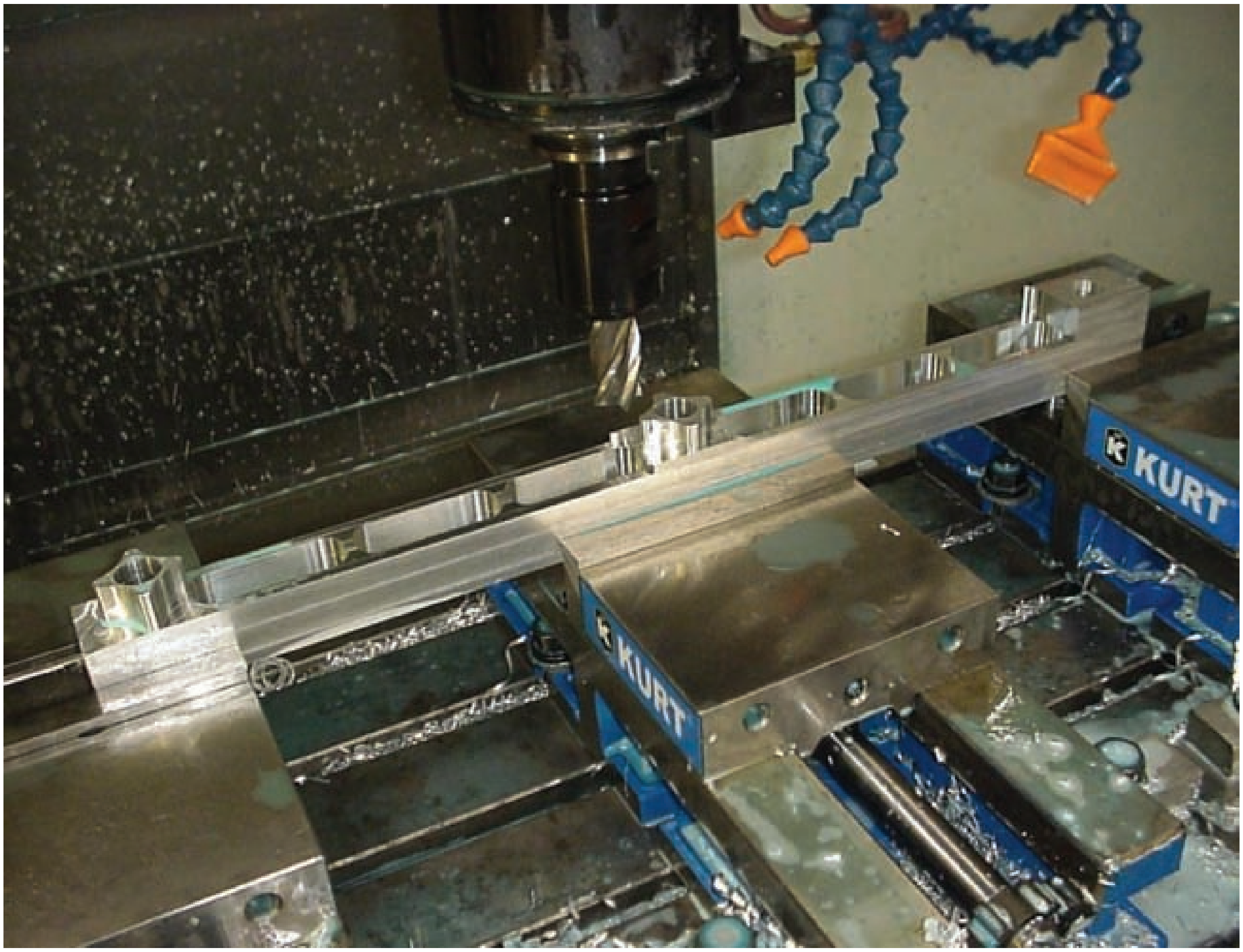


Figure 1-8 Using multiple matched vises is an easy way to hold long parts for machining.

7. Use multiple matched vises to hold long parts. (see Fig. 1-8)

Indicating multiple vises in line may take some time. However, once the vises are in place, they can usually be left alone for a while. Multiple vises can be used to run multiples of the same part using different starting points such as G54, G55, G56, etc. Unused vises can also be used to run other jobs to avoid disturbing a setup.

8. Determine how you are going to square the material. (see Fig. 1-9)

Squaring the five sides of a part that are accessible in a setup means that you are going to have to side mill four sides. The top surface can be face milled. Side milled surfaces are more prone to inaccuracies than face milled surfaces. An end mill used for side milling that is dull, chipped, or tapered will likely give you a poor, inaccurate surface. This is especially true when using longer end mills. Therefore, if you are going to side mill a surface and expect an accurate surface, use a fresh end mill.

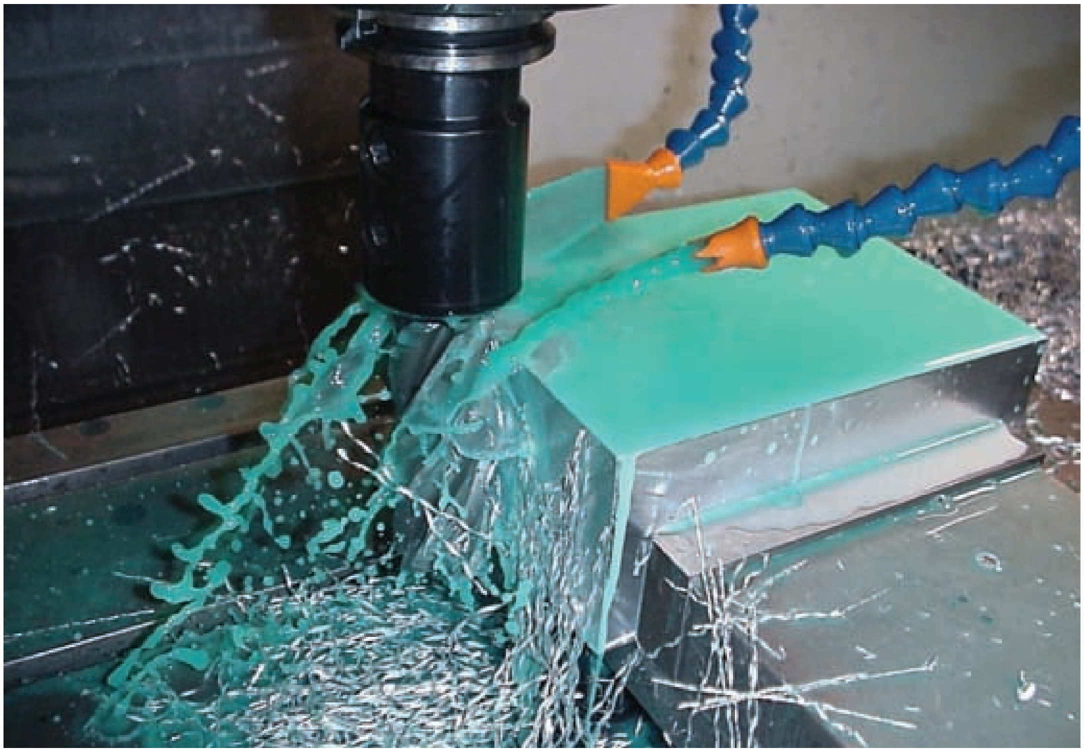


Figure 1-9 Side milled surfaces are more prone to inaccuracies than face milled surfaces, especially when using long end mills.

Figure 1-10 Small, stocky parts are less prone to warping than thin hogged-out parts.

9. Machine stocky rectangular parts in one setup. (see Fig. 1-10)

Small, stocky rectangular parts are less prone to warping and falling out of tolerance than hogged out parts. You can usually square them and do other machining operations in one setup. Using this method, you have to plan for holding excess stock in the vise. The excess material on the back side can be faced off later using a separate program.

10. Square larger parts independently. (see Fig. 1-11)

On larger, tight tolerance parts, I often do the squaring independently. In other words, I concentrate on getting good blanks first, then use other programs to add the necessary features.

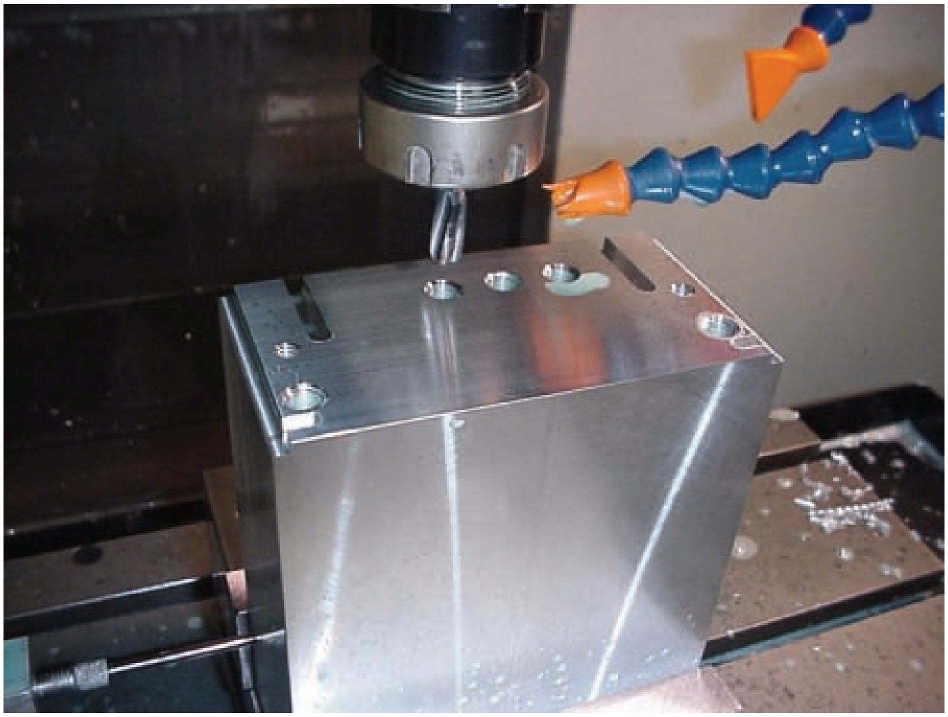


Figure 1-11 Sometimes it is wise to focus on getting precision blanks before adding features. The mold cavity above was precisely squared to size before doing any other machining operations.

11. Plan for material warpage.

Material warpage is a fact of life in machining; it is often an issue that must be addressed, especially with long, slender, and hogged-out parts. It may not be much of an issue on stocky, block-type parts or parts that need just a few simple features added.

One of the most effective ways to eliminate the tendency of material to warp is to cut through the skin of the stock material before doing other precision machining operations.

Occasionally, when a finished part has to be precisely square, parallel and on size, and has a lot of hogged-out features, you may have to rough in the large features first, then go "back" and complete the precision overall squaring and sizing of the part. Once the overall size is correct, you can add or finish close tolerance features such as dowel pin holes, knowing that the material will be stable. Using this technique is somewhat time consuming, but will likely produce parts that are in tolerance.

12. Make a few extra parts.

Running a few extra parts makes running a job faster and easier, I believe. When you have only one piece of material or the exact amount of material to complete an order, you necessarily have to be more careful — which usually translates into more time and stress.

13. Do a trial run on new programs

When making a first run test on a new program, note the changes you make at the machine for feeds and speeds. Also note any depth of cut and cutter path changes that would make the job run more efficiently. Until you watch a part being machined, you can't be certain what changes should be made.

After making notes at the machine, go back to the computer and update the settings or the cutting parameters in the CAM software; then re-output the program. Sometimes I do this a number of times, depending on how complicated or tricky a part is to run. The higher the quantity of parts, the more energy I put into this editing process. Ultimately, I like to have settings saved in my CAM software match what I am running at the machine so that very little editing, if any, is needed when the part is run at a later date.

You may have libraries of cutter feeds and speeds in your software. These are somewhat useful, but they don't account for the rigidity of the setup and the part you are machining. There's a lot of "feel" and intuition that goes into machining, whether it is manual machining or CNC machining. "Feel" is an acquired skill. It is sort of like hand tapping. If you tap enough holes, you eventually develop a feel for how much torque you can apply to a tap before it breaks.

Suggestions for Proper Planning

1. Initiate a new project with proper planning.
2. Use the largest contact area.
3. Rigidity is an important factor.
4. Spend time visualizing.
5. Determine if you can machine a job holding the material in a standard milling machine vise.
6. Face off the backside of a part to finish the overall thickness.
7. Use multiple matched vises to hold long parts.
8. Determine how you are going to square the material.
9. Machine stocky rectangular parts in one setup.
10. Square larger parts independently.
11. Plan for material warpage.
12. Make a few extra parts.
13. Do a trial run on new programs.

CHAPTER 2

Get A Grip

There are a lot of advantages to having a work piece held rigidly. When a part is held rigidly, feed rates can be increased and cutting time reduced. Cutters last longer and you'll likely end up with better surface finishes and more accurate parts. At "tool shows" where vendors offer impressive demonstrations, you rarely see flimsy, difficult-to-hold parts being machined. .

1. Hold parts securely and accurately. (see Fig. 2-1)

In real life, it's common to come across parts that are flimsy and difficult to hold. Holding parts for second operations such as drilling holes on edge can also be challenging.

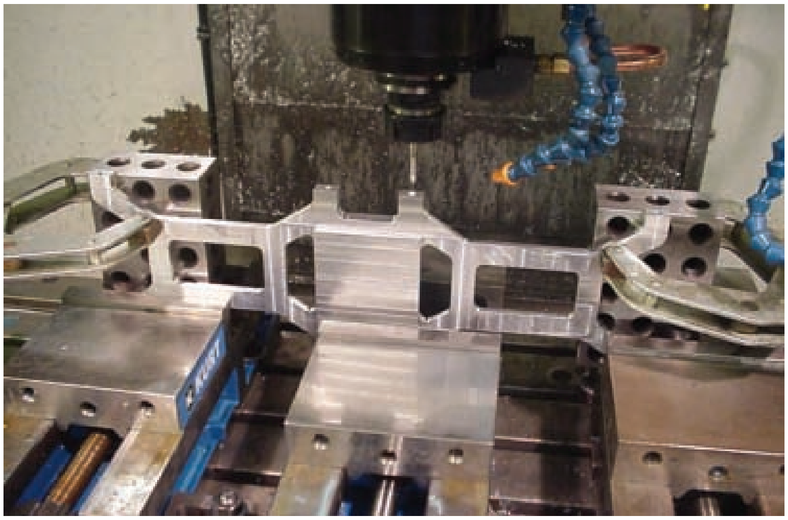


Figure 2-1 This part is being held for edge drilling with the aid of 2-4-6 blocks.

2. Find simple ways to hold parts securely. (see Fig. 2-2)

When presented with a part that is difficult to hold, your ingenuity will be put to the test. However, don't spend excessive time constructing complex fixtures when you don't have to.

The first options to consider when a standard vise won't do the job include using standard shop tooling — such as 1-2-3 blocks, 2-4-6 blocks, long parallels, grinding vices, angle plates, and V-blocks. You can often use these items effectively to provide the added support you need.

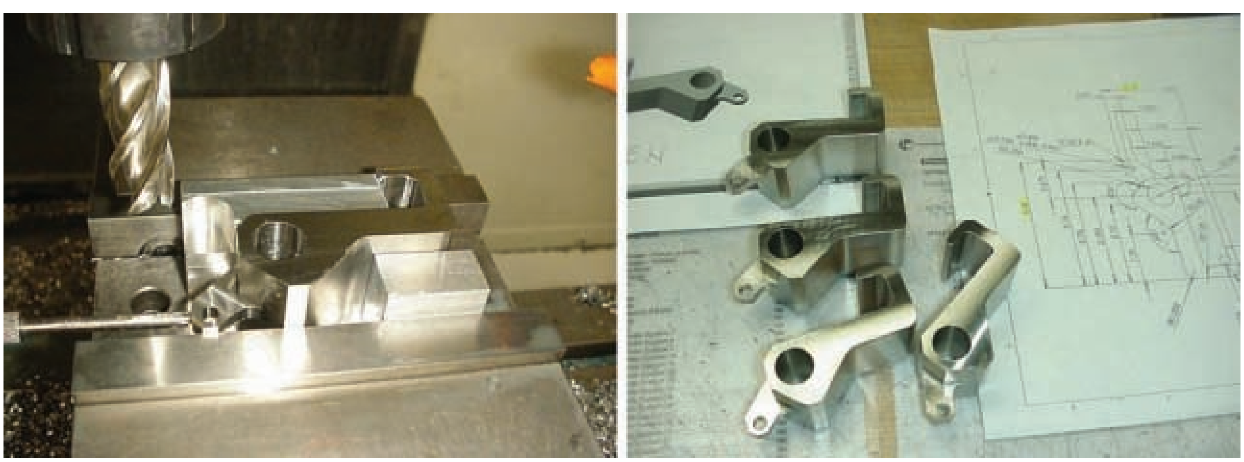


Figure 2-2 Some parts are difficult to hold. This part was ultimately machined using a standard milling machine vise in conjunction with a couple of spacer blocks.

3. Make a fixture, if needed. (see Fig. 2-3)

I've found that when you finally decide you need a fixture and take the time to make one, you're almost always glad you did. Often a fixture is needed to hold parts for perimeter cutting if the material is already to the correct thickness. In these cases, you can't use excess stock to hold the part for perimeter cutting.

Figure 2-3 A 2-4-6 block in combination with a gauge block stack is used to support the center section of a bar so it doesn't "push away" during the cut..

4. At the end of a CNC program that drills holes in the part, add a tap drill to drill deeper into the mounting fixture. (see Fig. 2-4)

Here's a simple but effective trick for making fixtures you can bolt parts to. Suppose you are making a part that has no excess material to hold for cutting the perimeter, but has through holes that can be used for clamping. When you are done with the drilling program, hand tap the fixture through the holes in the part. After that, you simply screw the part to the fixture. You are now ready to cut the perimeter. The beauty of this method is you never have to move the part.

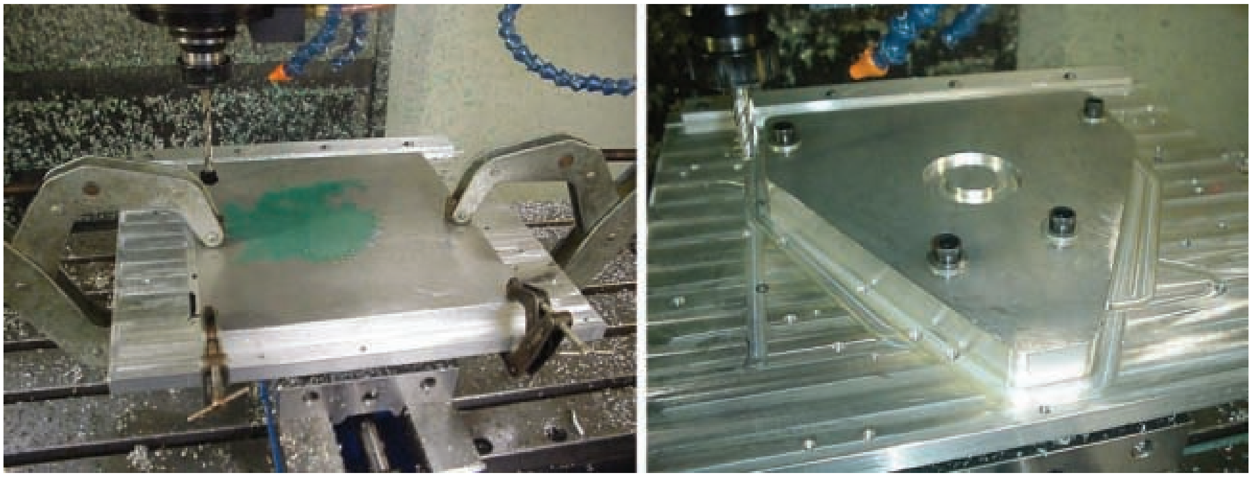


Figure 2-4 A tap drill can be added to the part drilling program to drill into a fixture plate.

5. Save fixtures for future use. (see Fig. 2-5)

Most fixtures can be constructed using aluminum. We have a couple of drawers full of simple fixtures that machinists have constructed over the years for various jobs. One of our biggest problems with fixtures is that we don't have enough room to store them. Therefore they get thrown here and there, scattered around, which makes them difficult to find.

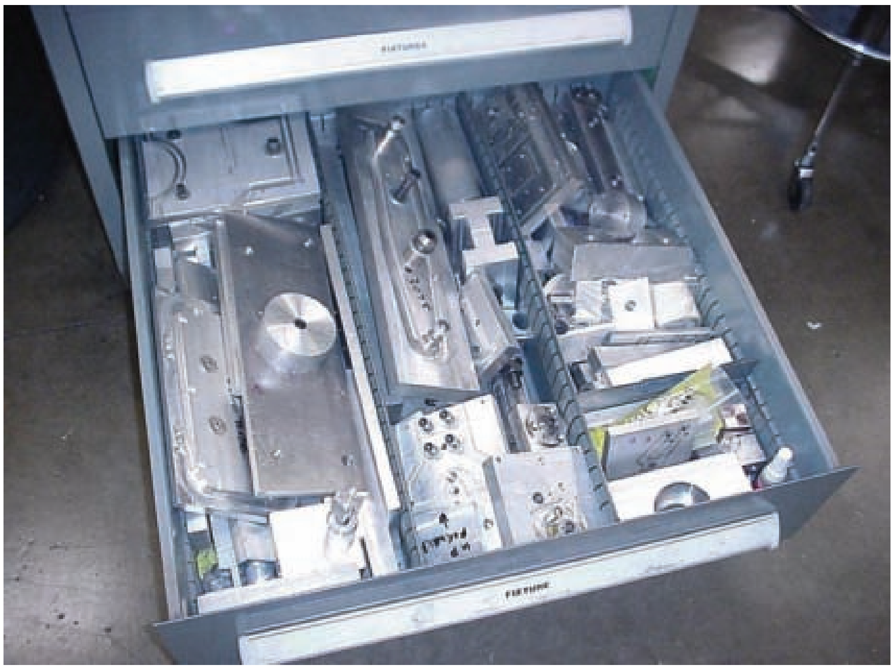


Figure 2-5 Fixtures become worth their weight in gold when you need them.

Although sheet metal parts look simple enough, they are usually a pain to hold and machine. I dislike seeing guards, housings, panels, electrical boxes, etc. come through the door simply because they are flimsy and difficult to hold (see Fig. 2-6).

You're often forced to make elaborate, hare-brained, time-consuming setups to hold these parts just so you can machine a few simple slots, holes, or windows in them. You may find tips 6 through 12 to be beneficial; they are all related to work holding.

6. Use a good quality angle lock milling machine vise.

We made the mistake some years ago buying a couple of cheap imported milling machine vises. What a blunder! Our thinking at the time was: "All the vise has to do is clamp the part. How difficult can that be?"

We suffered with these vises for about a year before we upgraded to high-quality vises. Table 2-1 shows what we found about using cheap milling machine vises.

A milling machine vise is something used nearly every day so the benefits of having high quality vises are compounded.

Cheap milling machine vises

- You are constantly struggling to keep parts held securely against parallels.
- There is a "spongy" feel when you start clamping on a part. You're never quite sure how much to tighten the vice, which results in inconsistent clamping pressure and part location.
- They have a lot of sharp edges and burrs.
- They are constructed of cheap metal; tolerances are wide open; even the handles don't fit well.

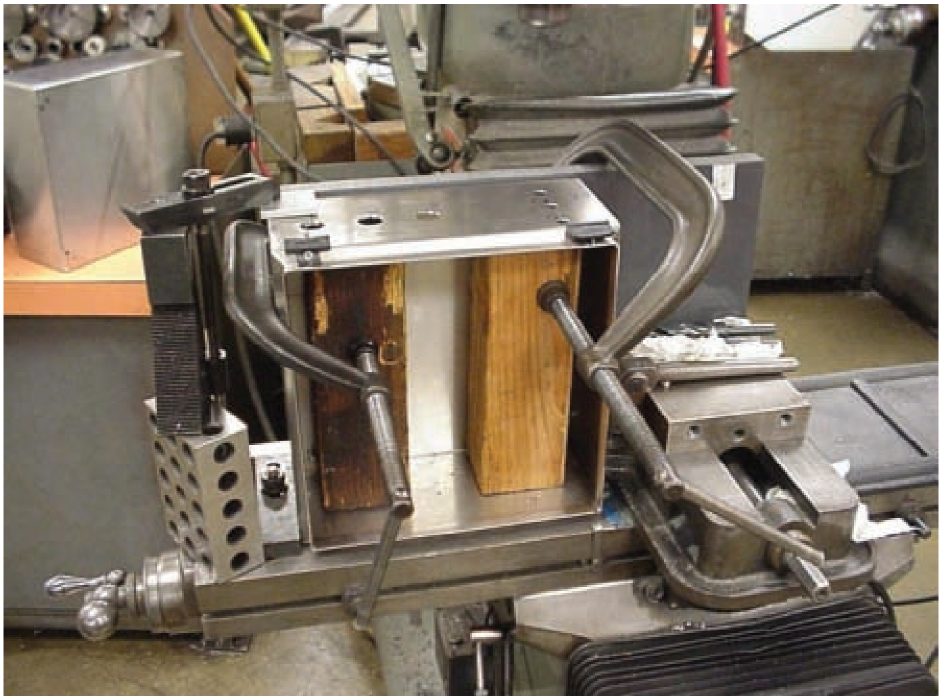


Figure 2-6 Sheet metal parts are often difficult to hold for further machining.

7. Plan setups so you can see the cut. (see Fig. 2-7)

This is a rule I always try to follow. There is no sense working blind if you don't have to. Being able to see the cut has advantages: You'll be able to keep an eye on cutters for anomalies such as chip packing, chatter, coolant coverage, surface finish, cutter flexing, gouging, jamming, and anything else that might become an issue. As an added benefit, features you cut will likely be easier to inspect.

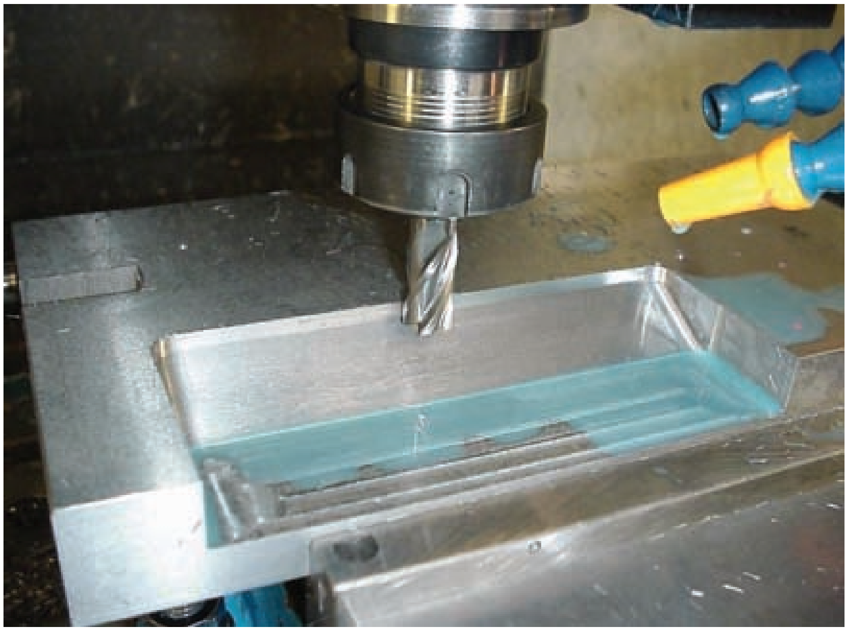


Figure 2-7 The programming and setup for this cut were planned in advance so the cut would be facing the operator. Being able to see the cut makes it easier to watch for programming and cutting anomalies.

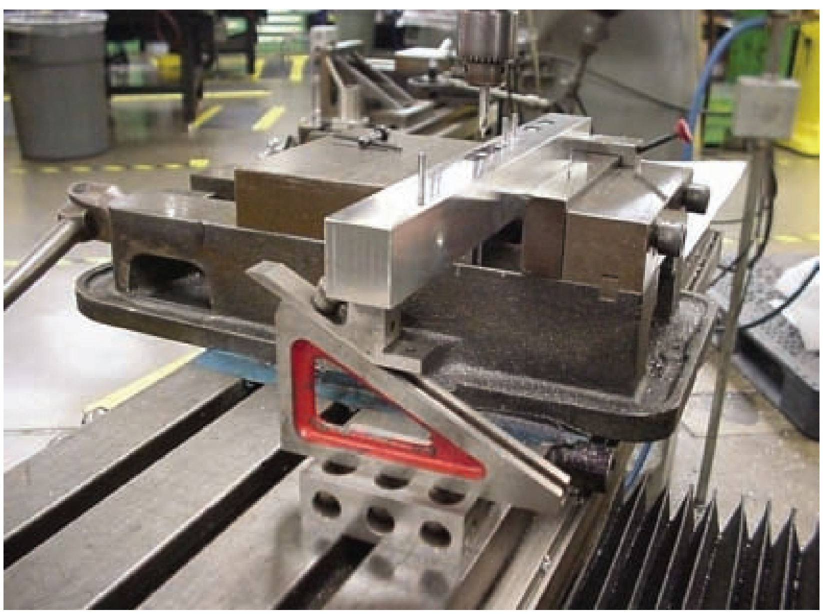


Figure 2-8 A planer gage is used to quickly add stability to this setup.

8. Aim for stability in your setups. (see Fig. 2-8)

Theoretically, you have stability when pressure applied anywhere perpendicular to an unclamped part doesn't move the part. Having stability is more important when machining flimsy parts —the strength of the material may not provide enough rigidity to make a cut.

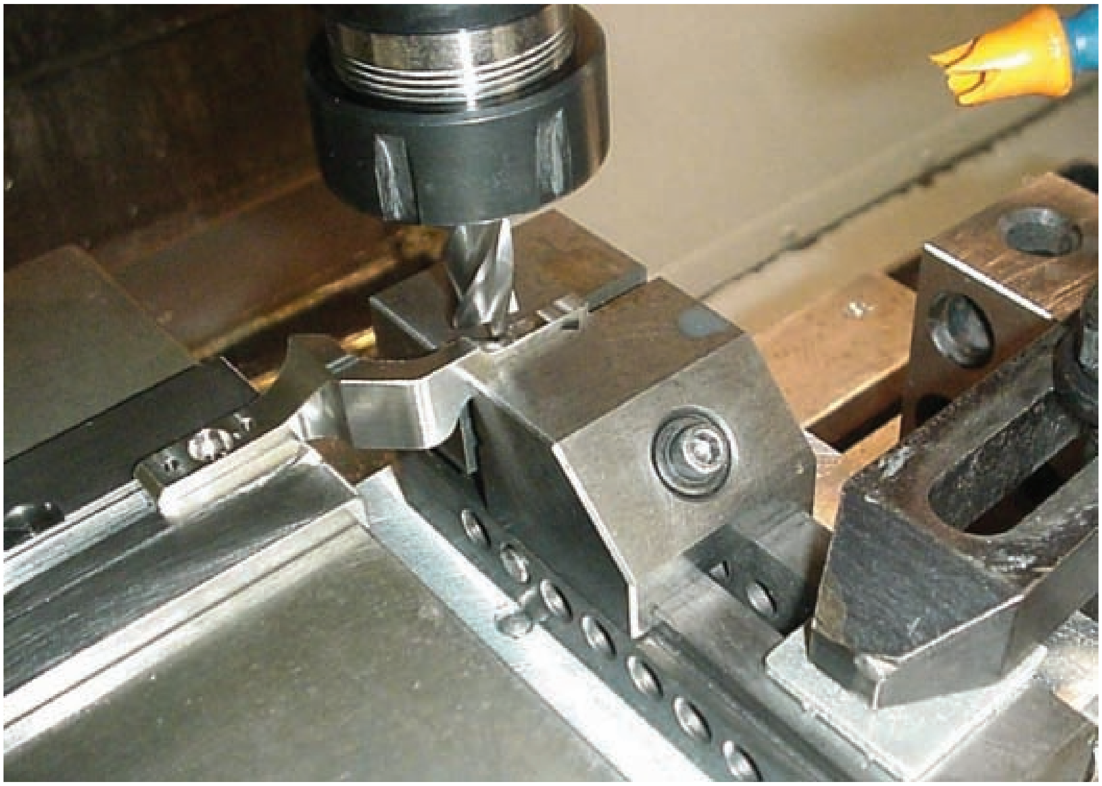


Figure 2-9 Dog leg parts such as this can often be given additional support by mounting a temporary vise to the table.

9. Use a grinding vise for additional support. (see Fig. 2-9)

This arrangement works great for supporting odd-shaped parts that can't be held conveniently in the main vise.

10. Construct a set of long jaws that can be bolted to the vise. (see Fig. 2-10)

Cut a groove down the middle of two aluminum bars mounted side by side to create a set of long jaws. We have a few sets of these parallels that have come in surprisingly handy over the years. Long jaws that extend beyond the width of the vice will not provide the same clamping pressure at the ends of the jaws as in the middle. Use a couple of Kant-Twist clamps at the ends of the jaws to provide adequate clamping pressure to hold parts firmly.

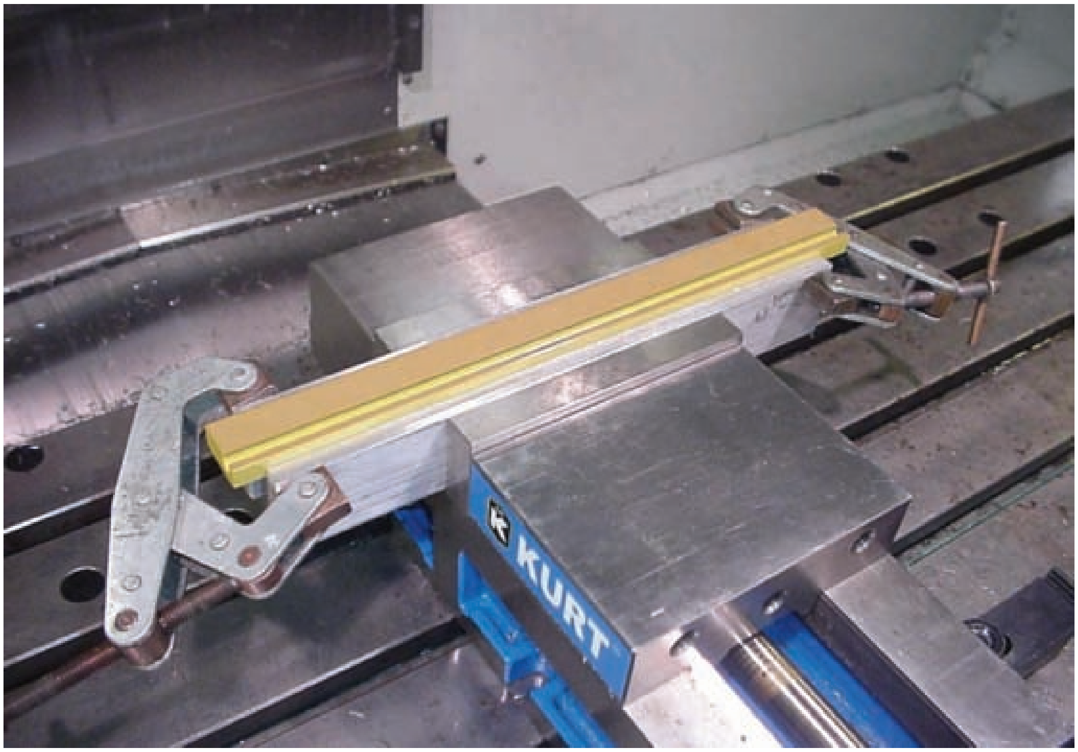


Figure 2-10 Long jaws such as these come in handy for holding parts that extend beyond the vise jaws.

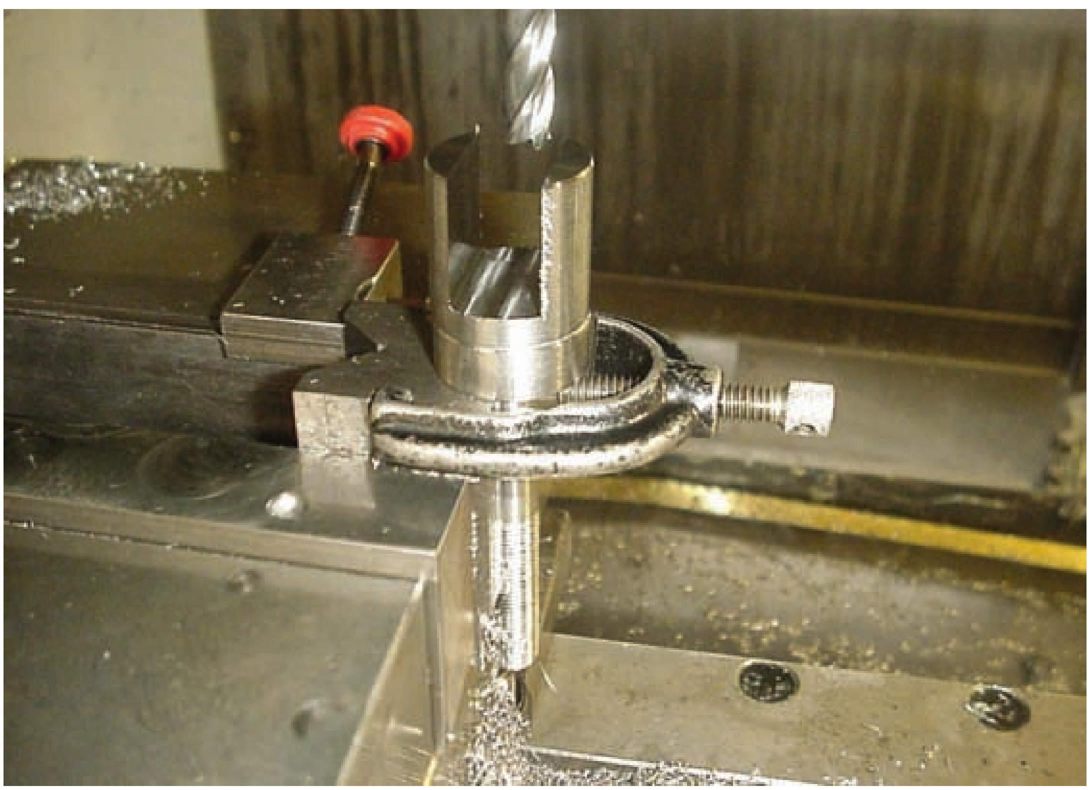


Figure 2-11 V-blocks provide a quick way to hold cylindrical parts for further machining.

11. Use V-blocks to hold round parts. (see Fig. 2-11 and Fig. 2-12)

There are a few issues to keep in mind when using V-blocks to hold parts for machining. First, V-blocks don't apply clamping pressure over a large surface area. The clamping pressure is applied to tangent lines on the circumference of the part. If you squeeze down too hard on parts held with V-blocks, you may dent the parts and throw the center line of the part off location.

If you use two V-blocks to hold round parts, theoretically you get four lines of clamping contact. If I know cutting pressures will be light, I often use just one V-block and the vise jaw to apply clamping pressure. You can also use some soft material between the vise jaw and the part to avoid denting the part.

If the diameters of cylindrical parts are not precise and consistant, using a V-block to hold parts may cause them to tilt or may throw center lines off location. That's one reason you shouldn't always take advantage of wide-open tolerances. Tight tolerances, although not necessarily needed for the part to function, may be needed to locate and hold parts accurately for further machining.

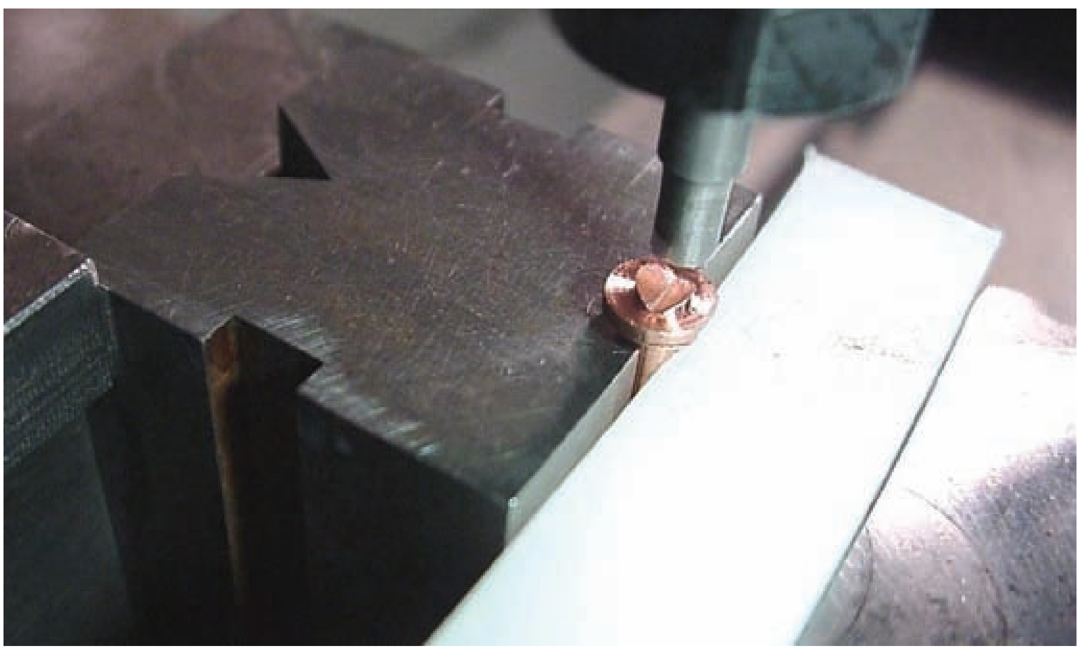


Figure 2-12 This V-block has a relatively small "V" groove cut which is used to hold a small copper pin for additional machining. Plastic material is being used here to apply pressure to avoid denting the part.

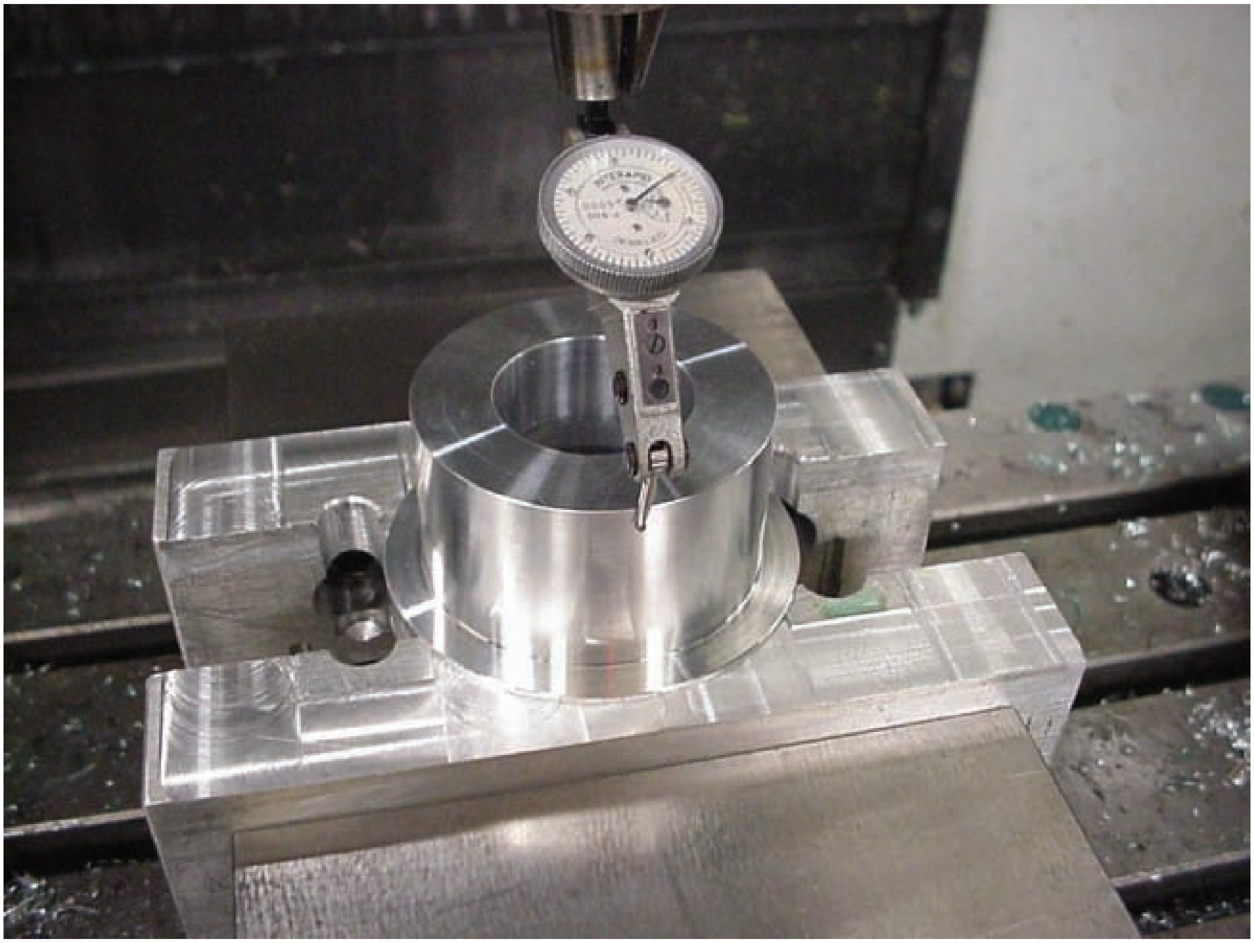


Figure 2-13 Perishable vise jaws can be used effectively to hold parts that may be difficult to hold using other methods.

12. Use perishable vise jaws. (see Fig. 2-13)

Use perishable vise jaws as a last but effective method to hold parts. I use perishable vise jaws as a last resort for the following reasons. It's often difficult to find a set that hasn't been all cut up. You can always make new vise blanks, but if you leave them out, they'll likely get used quickly. Furthermore, the appropriate nests have to be machined into them, which takes extra time.

Suggestions for Get a Grip

1. Hold parts securely and accurately.
2. Find simple ways to hold parts securely.
3. Make a fixture, if needed.
4. At the end of a CNC program that drills holes into the part, add a tap drill to drill deeper into the mounting fixture.
5. Save fixtures for future use.
6. Use a good quality angle lock milling machine vise.
7. Plan setups so you can see the cut.
8. Aim for stability in your setups.
9. Use a grinding vise for additional support.
10. Construct a set of long jaws that can be bolted to the vise.
11. Use V-blocks to hold round parts.
12. Use perishable vise jaws.

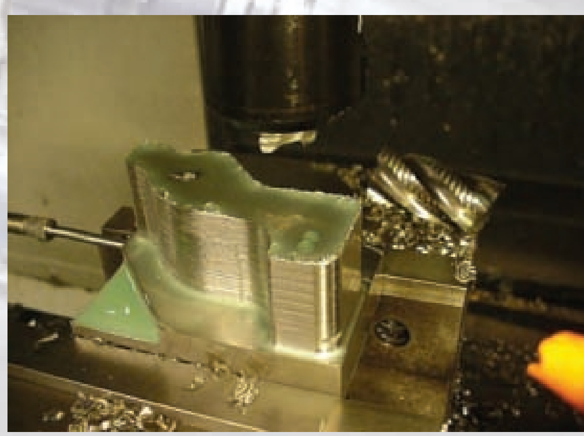
CHAPTER 3

Avoiding Crashes

Everybody crashes, I was once told. It's likely true. However, not all crashes are created equal. There are fender benders that may just break small tools, and then there are head on collisions that may ruin your day.

Crashes are relatively easy to avoid in CNC machines. They often occur during setup and debugging. If you can recognize high-risk situations, you'll be in a better position to avoid them.

Before discussing the mechanics of avoiding crashes, I'd like to highlight something many of you already know: One of the most effective ways of avoiding crashes is to avoid being interrupted while programming and setting up. That's easier said than done, of course.



Having said that, this chapter provides a set of tips that have saved me numerous times!

1. Avoid moving around too much in Handle Jog mode.

If you are not familiar with the term, Handle Jog mode allows you to move the table around manually with a CNC machine. I don't like to do much of anything with a CNC machine in Handle Jog mode. I use Handle Jog mode mostly for the bare necessities such as edge finding, indicating, and clearing the cutter. However, Handle Jog mode can be used for some simple machining.

Occasionally I use Handle Jog mode to face the end of a bar or fly cut blocks of material. However, I generally prefer machining parts under program control. We're only human and it is relatively easy to forget which axis and feed increment you have engaged when you start cranking the feed handle. A good habit to get into is to be cautious when you start cranking the feed handle. Turn the handle just one or two clicks to verify that the spindle is moving in the direction and feed rate you want.

2. Before running a new program, scan through the program at the machine to check for gross errors. (see Fig. 3-1)

Check the Z negative moves in a new program to see if they make sense. This is an easy and often fruitful way to avoid crashes. For example, if your first tool is a center drill and you are drilling a plate, then you know the Z negative move for the center drill should be somewhere around Z –.150. If the value in the program is something like Z –1.150, then you know right away something is wrong.

You should also scan other Z negative moves in a program for drilling, reaming, tapping, etc. The Z negative moves for those operations can be easily found within the canned cycles that execute them.

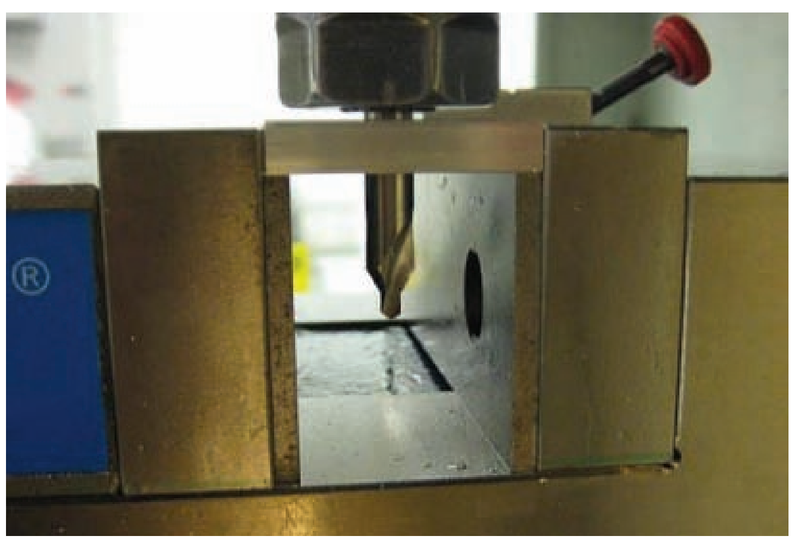


Figure 3-1 Gross Z negative errors in a program translate into gross machining errors. In this example, the center drill was mistakenly programmed to go Z–1. instead of Z–.1, which resulted in a scrapped part. Machinists should do a visual scan of a new program to check for these types of errors.

3. Perform a quick visual scan of feed rates.

If you see something like F500., and you are machining stainless, then you know right away that something is wrong, at least with the feed rate. Feed rates for machining stainless are generally in the F10. to F20. range. F500. would certainly break a cutter.

4. Perform a quick visual scan of spindle speeds. (see Fig. 3-2)

Spindle speeds are sometimes incorrectly input in a program. It is easy for a programmer to add another zero to a spindle speed by mistake. Suppose a spindle speed for a reamer was meant to be 300 RPM and the programmer adds another zero and makes it 3000 RPM. In this case, the reamer is going to get fried if you don't catch the mistake.

Before running a program, also make sure that all tool numbers are correct and that they have the correct corresponding "H" value. (The H value calls up the tool length offset for a specific tool.) For example, in a section of programming for a specific tool, you would not want to see "T2 M06" followed by "G43 H4." The correct programming would be "T2 M06" followed by "G43 H2." Only after calling up T4 would you want to see G43 H4 in a program.

I've gotten to the point where I shy away from letting other machinists run my unproven programs because a lot of them won't take the time to do these simple checks. Further discussion of programming code will follow in Chapter 8.

Figure 3-2 It's a good idea for machinists to physically lay out all the tools used in a program in sequential order before installing them in a machine. Most tool numbering errors can be caught early in this way.

No programmer can provide perfect programs all the time. I'll even go so far as to say that if I'm the machinist, and I run your program, which results in a crash, it's my fault. In other words, with few exceptions, I believe the machinist has the responsibility and the means, within reason, to make sure everything is going to run OK.

Tool numbers get screwed up in programs for various reasons. During program construction, programmers have to decide what tools to use. Tools often get added or subtracted by the programmer. If the programmer fails to renumber the final tool selections in a program, confusion can occur while setting up the job.

5. Reduce the "Rapid" speed (G00) during setup and debugging by activating a lower rapid speed percentage button.

Reducing the rapid speed is a precaution that gives you more time to hit the feed hold button if something doesn't look right. I often use these lower percentages during setup and debugging, especially when I am working on expensive parts.

6. For first runs and debugging, toggle between the "Cycle Start" button and the "Feed Hold" button on the controller as the cutter approaches the part. (see Fig. 3-3)

It is difficult to see where a cutter is in relation to a part when the spindle is at Machine Z Zero (retracted). The closer the cutter gets to the part, the easier it is to see the relationship between the part and cutter. Sometimes I toggle half a dozen times when a cutter is on its way down, especially if I'm doing something like engraving expensive mold cavities.

Figure 3-3 The machinist is being careful here by toggling between the cycle start button and the feed hold button as the cutter approaches the part. If there is a programming error, the machinist is in a good position either to catch the error before any material is cut or to reduce the damage the programming error may cause.

7. Insert M1 "Optional Stop" commands in the program before each tool change. (See Fig. 3-4)

For first runs and debugging, it is useful to have the "Optional Stop" activated so that you can be at the machine for the start of a new tool. Often, the first few moves a new tool makes will let you know if there is a problem. Once you determine all tools are running okay, you can deactivate the "Optional Stop" button and start running.

8. Clamp large remnants in place or remove them so they can't fall behind the machine table. (see Fig. 3-5)

One time I screwed up one of the sheet metal covers that cover the back column. I failed to remove a remnant that had fallen behind the machine table. When the table moved toward the column while a program was running, the remnant got jammed between the machine table and the cover, which destroyed the cover. Lesson learned.

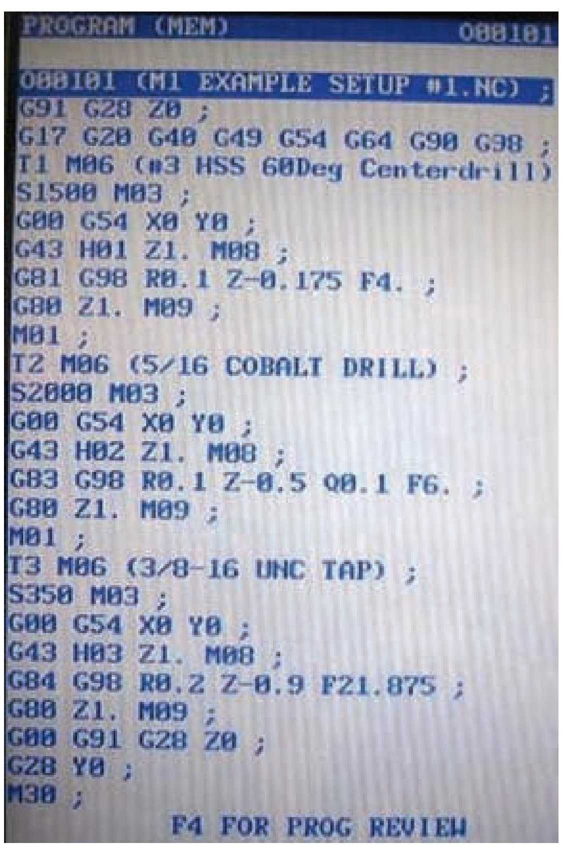


Figure 3-4 This is a short program showing M01 Optional Stop commands inserted in the code before the tool change commands T2 M06 and T3 M06. If the Optional Stop button on the controller is activated, the machine will stop executing the program when the controller reads the M01 commands. Optional Stops are generally used for debugging and proofing programs because they give the machinist a chance to see how the last tool ran.

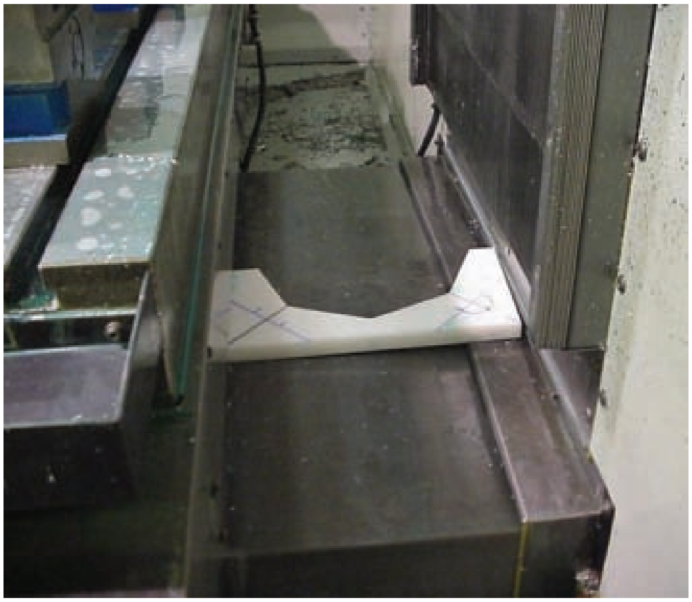


Figure 3-5 Large remnants can get jammed between the table and the back column of the machine. It's a condition that must be avoided.

9. Program drilled pilot holes deep enough so reamers won't jam.

The pilot hole for a reamer has to be drilled deep enough so that the reamer won't jam against the bottom of the pilot hole. When you have blind reamed holes, it is a good idea to check the program at the machine to make sure pilot holes are drilled deep enough to avoid this interference.

10. Be extra cautious when using tie-down clamps. (see Fig. 3-6)

Except for simple setups, I try to avoid using tie-down clamps unless I know there is little chance of running into them. Besides just sticking up in harms way, they are too often placed inconsistently by operators. Instead of using tie-down clamps, I prefer bolting parts down. I try to avoid special programming that may be needed to "jump" over clamps. These special programs can be nightmares for someone trying to set up the job at a later date.

11. Be cautious of protruding bolt heads. (see Fig. 3-7)

With bolts, a default rapid plane of one inch is usually adequate to jump from one location to the next without having to concern yourself with interference — as long as the bolt heads don't stick up higher than the rapid plane.

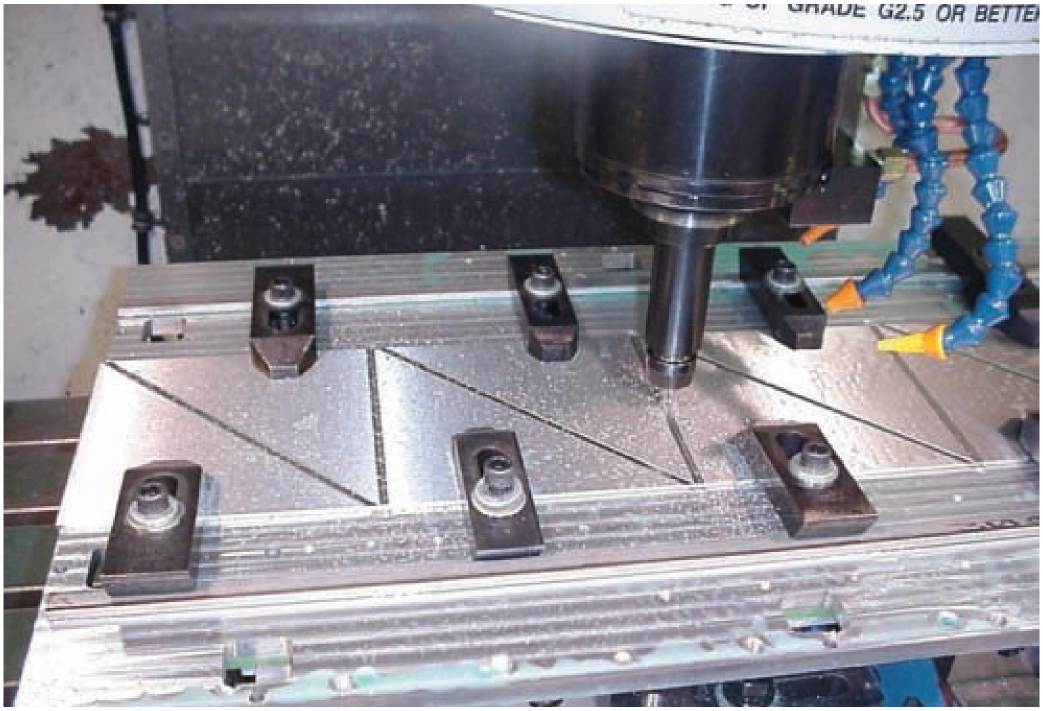


Figure 3-6 Tie-down clamps are something that can easily be run into. When using tie-down clamps, program using generous rapid planes.

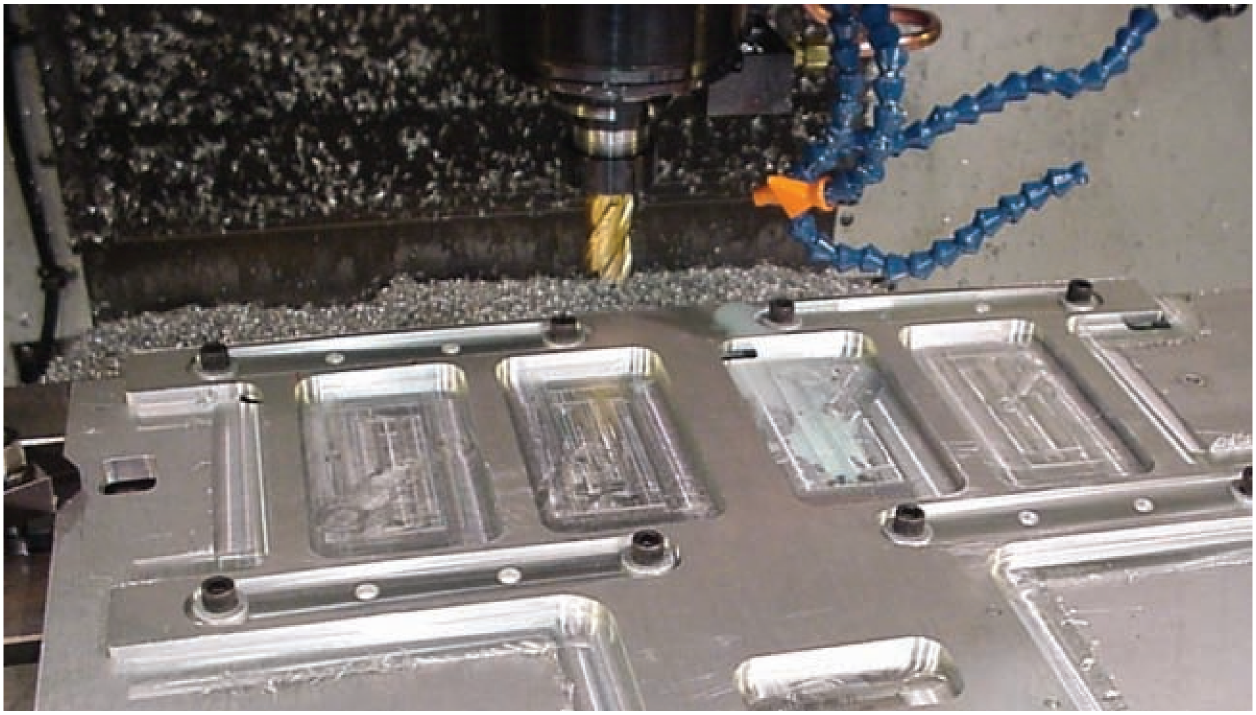


Figure 3-7 Bolting parts down for machining is preferable to using tie-down clamps because bolt heads don't stick up as high and are used in consistent locations.

12. If a tap breaks while you're at the machine, hit either the "Emergency Stop" button or the "Reset" button.

Once the machine starts executing a G84 tapping canned cycle, pressing the "Feed Hold" button will not stop the spindle from continuing its journey to tap the hole. This can make a bad situation worse as the broken tap, which is still in the tool holder, grinds into the portion of the tap that is broken off in the part.

13. When mounting cutters in tool holders, make sure all cutters extend far enough below the tool holder so the holder doesn't interfere with the part when the program is run. (see Fig. 3-8)

This is a common crash, one that can easily be avoided if the machinist takes a little time to understand what features a tool is going to cut and how deep the cutter will be going. Most setup sheets provide this information.

I would be a nervous wreck running somebody's unproven program until I fully understood what feature a tool was going to cut. Sometimes cutter depths are obvious by looking at the drawing. Other times — especially when there are many tools and the part is complicated — it's not so obvious. If you have the option and the experience, it is best to check how deep the cutters will be going by referring to the CAD model and the CAM simulation.

Standard center drills are relatively short. Be especially careful with standard center drills in order to avoid tool holder interference. You can purchase long center drills; they work well for reaching down to lower levels of a part.

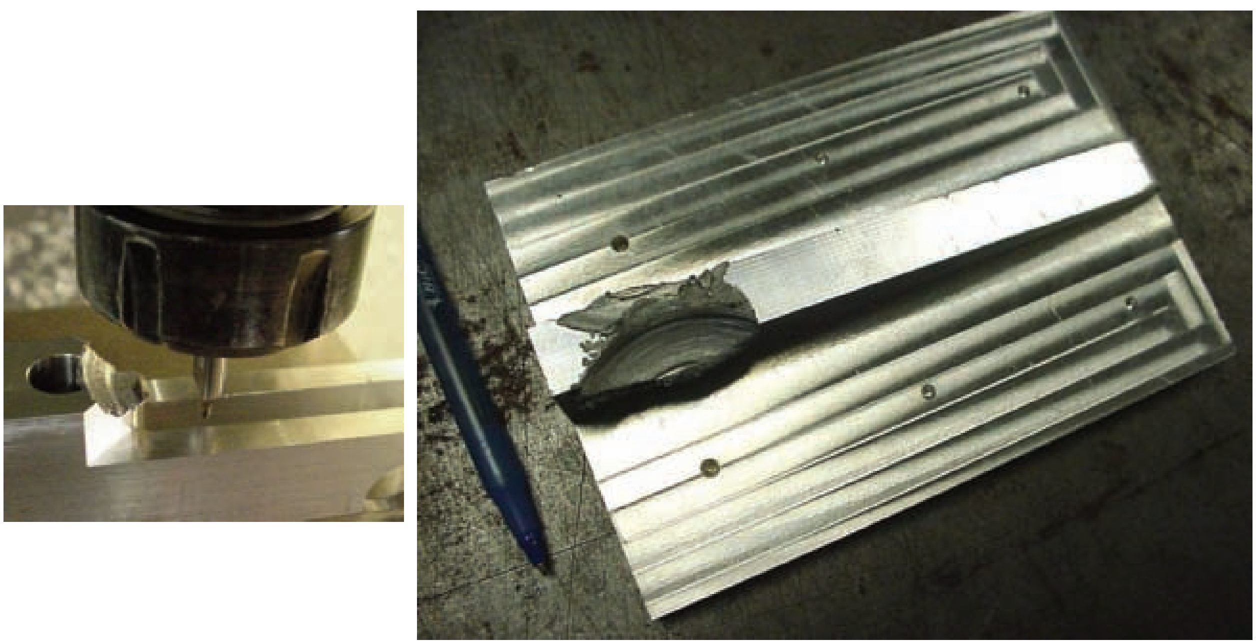


Figure 3-8 Tool holder interference is a common crash. This type of crash can easily be avoided if the machinist knows what the tool is going to cut and also how deep it will cut.

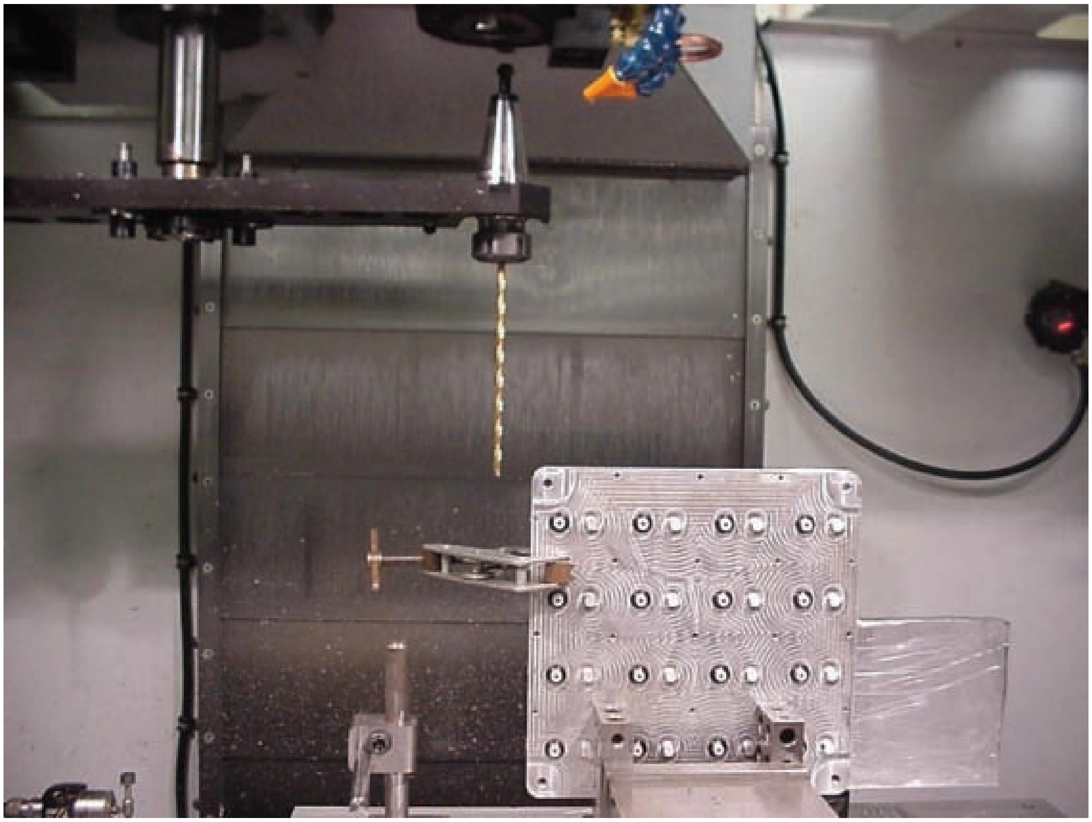


Figure 3-9 Long tools and tall parts are a recipe for crashes during tool changes. In this case, an edit to the program was necessary to move the part out of the way before conducting a tool change.

14. Manually edit programs to avoid crashing long tools into tall parts. (see Fig. 3-9)

If a tool extends below the top of the part at any time during a tool change, be careful with both your programming and setup. You want to avoid crashing the tool into the part during a tool change. If you are careful, it is possible to work around any interference that could happen. Remember to move the part out of the way before manually cycling the tool changer. Then, before running the program, you may need to edit the program manually to move the part out of the way before a tool change.

An example of an edit that moves the part out of the way before a tool change is as follows:

G00 X-10.

(Moves the part 10" and presumably out of the way before the tool change)

15. Set remnants free by switching to a smaller diameter end mill to cut through the last few thousandths in Z. (see Fig. 3-10)

When remnants start coming loose from a part, it's often time for concern. In my experience, the behavior of a remnant coming loose is unpredictable. Remnants coming loose can cause jamming.

If the remnant is in an enclosed space, and you only use one size end mill to cut through the part, the chances of the remnant jamming against the cutter are high.

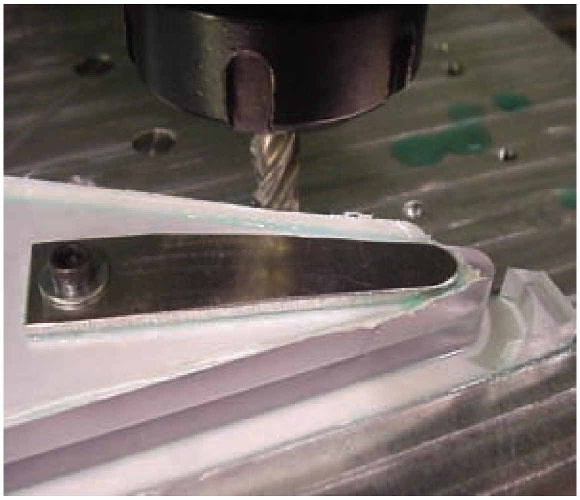


Figure 3-10 When the corner remnant on the right starts coming loose, it could very well cause jamming. To be safe, a smaller diameter end mill should be used to cut it free.

16. There are various ways of dealing with the jamming remnant issue. (see Fig. 3-11)

The easiest way is simply to convert everything you don't want into chips. If I have just a few parts to make, I'll write a program to do just that. If the part has large windows or the material is tough to cut, the time needed to turn unwanted window material into chips can be significant — you may want to choose another method.

If you can clamp the remnant in place with either clamps or bolts, that will solve the jamming issue. If clamping the remnant is impractical, another way to reduce jamming is first to use a larger diameter end mill to cut through the majority of the material in Z, then switch to a smaller diameter end mill to cut through the last few thousandths to set the remnant free.

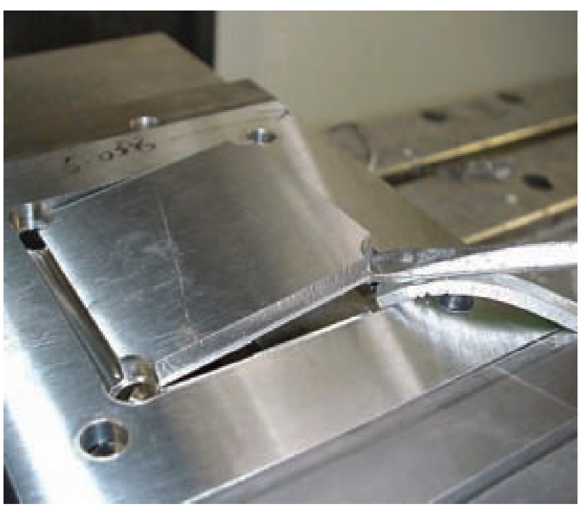


Figure 3-11 I'm cheating here by manually removing the remnant. In hind sight, I could have used a smaller diameter end mill to run around the window to cut the remnant free.

Suggestions for Avoiding Crashes

1. Avoid moving around too much in Handle Jog mode.
2. Before running a new program, scan through the program at the machine to check for gross errors.
3. Perform a quick visual scan of feed rates.
4. Perform a quick visual scan of spindle speeds.
5. Reduce the “Rapid” speed (G00) during setup and debugging by activating a lower rapid speed percentage button.
6. For first runs and debugging, toggle between the “Cycle Start” button and the “Feed hold” button on the controller as the cutter approaches the part.
7. Insert M1 “Optional Stop” commands in the program before each tool change.
8. Clamp large remnants in place or remove them so they can’t fall behind the machine table.
9. Program drilled pilot holes deep enough so reamers won’t jam.
10. Be extra cautious when using tie-down clamps.
11. Be cautious of protruding bolt heads.
12. If a tap breaks while you’re at the machine, hit either the “Emergency Stop” button or the “Reset” button.
13. When mounting cutters in tool holders, make sure all cutters extend far enough below the tool holder so the holder doesn’t interfere with the part when the program is being run.
14. Manually edit programs to avoid crashing long tools into tall parts.
15. Set remnants free by switching to a smaller diameter end mill to cut through the last few thousandths in Z.
16. There are various ways of dealing with the jamming remnant issue

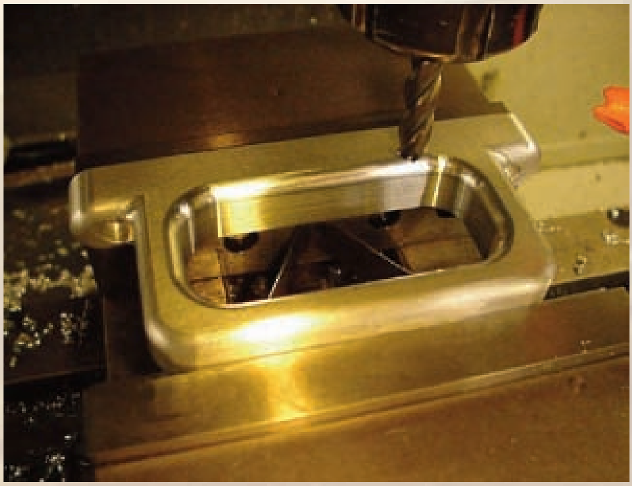
CHAPTER 4

Chamfering and Edge Dressing

The expression "There is more than one way to skin a cat" applies well to machining. It is common that once you make a program, somebody will suggest the job should be run a different way. That's one reason I prefer doing everything myself!

I don't think it matters how anybody runs a job as long as the final product is acceptable, not too many cutters were destroyed in the process, and the work is completed in a reasonable amount of time.

We have a young programmer in our shop who is constantly being run in circles by the whims of machinists and other people in authority. I feel a bit sorry for him. Newcomers are in a difficult position. Not only do they lack experience, but they generally want to please everybody.



There are a number of ways to apply edge dressings to parts — some work better than others. In this chapter, we'll explore ways that may make it easier for you to apply these common features.

1. Chamfering is not as easy as it looks. (see Figs. 4-1 and 4-2)

Chamfering looks easy on a computer (Fig. 4-1). But, in reality, it can be a little tricky to cut a clean chamfer (Fig. 4-2).

The point of a precisely sharp chamfering tool does not cut well and will throw a burr if programmed to cut on center around the profile of a part. The reasons for this are a) not much chip clearance available at the end of a pointed chamfering tool and b) surface feet per minute at the tip of a pointed tool theoretically goes to zero. Also, the tip of the tool may not be precisely sharp.

Figure 4-1 This simulated chamfering tool is running with its tip on the edge of the part. It looks great — on a computer!

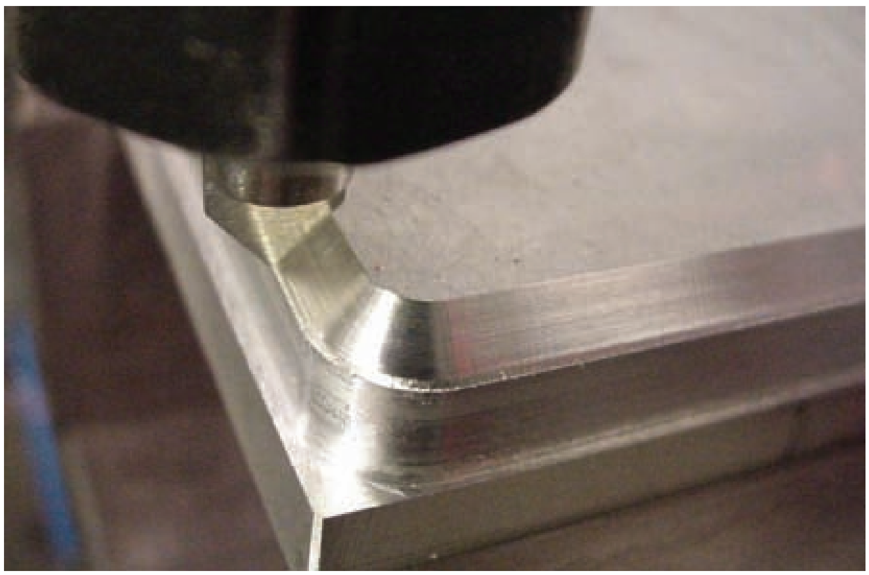


Figure 4-2 This chamfering tool has the center of the cutter runnning on the edge of the part. The chamfer doesn't look as great now.

2. For a chamfering tool to cut cleanly, the tip of the tool must be offset from the profile. (see Figs. 4-3)

The tip of the tool must be outside the profile (Fig. 4-3). I noticed recently in a new version of the CAM software we use that you can now offset a chamfering tool to cut a clean chamfer. Offsetting a chamfer tool is the only way to cut a clean surface.

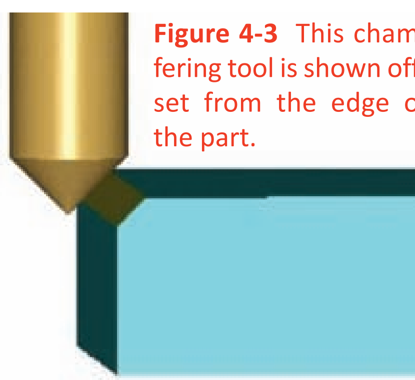


Figure 4-3 This chamfering tool is shown offset from the edge of the part.

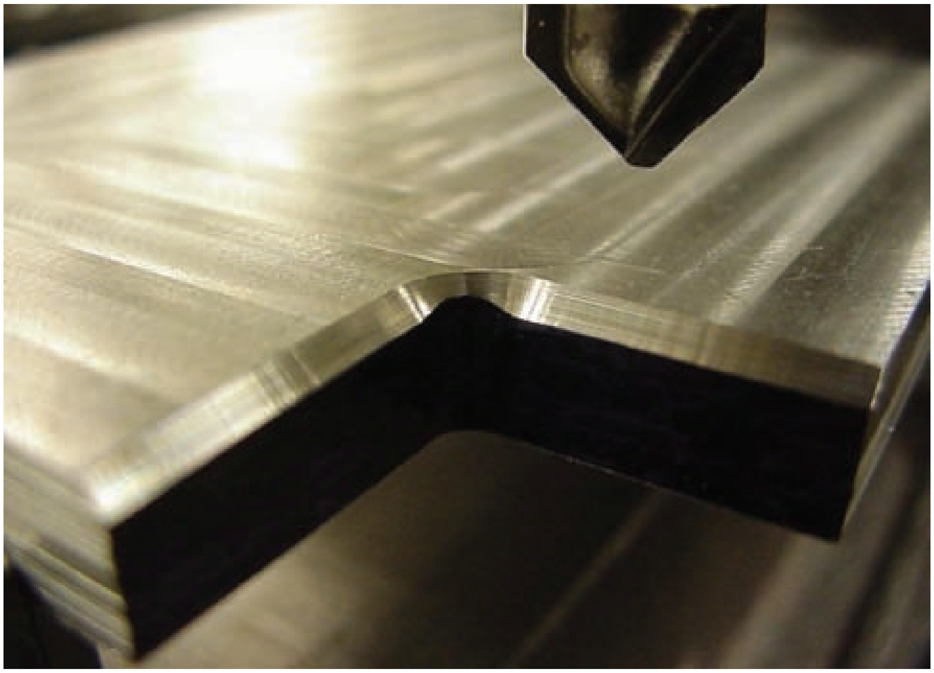


Figure 4-4 This example shows an inconsistent chamfer on an inside radius. The Z height of the chamfering tool wasn't set correctly.

3. If you offset a chamfering tool to cut cleanly, the depth of the tool must be correct. (see Fig. 4-4)

If the depth is not correct, the chamfer can produce inconsistent chamfers on inside radii. In Figure 4-4, the chamfer is consistent except on the inside radius. The depth of the chamfering tool was not set properly to match the inside radius.

These issues can be overcome with thoughtful planning. However, when it boils down to getting things down, especially for such trivial features, I don't like to think too much. I prefer to let the computer do the thinking…so I cheat!

4. For most chamfering, use a ball nose or hog nose end mill, and program the chamfer using a 3D cut. (see Fig. 4-5)

Using this method, the computer does all the thinking for you and you don't have to worry about burrs, offsetting the tool, or inconsistent chamfers.

The only thing you have to concern yourself with is choosing a ball nose or hog nose cutter with a radius smaller than the smallest inside radius of the part. The downside is that chamfers may take a little longer to cut because you are taking multiple passes to create the chamfer — but not much longer. I usually go full throttle when chamfering this way. Be aware that using this method may raise some eyebrows.

The young programmer mentioned earlier got into hot water once using this method. He was making the part shown in Figure 4-5. The part has a .1″ chamfer around the perimeter and the center window. The customer was waiting for the part and would frequently walk over to the machine to check its progress. The chamfer was going painfully slow and the customer was getting agitated. I overheard something like “why don’t you just file the damn thing or grind it or something?”

Afterward, I consulted with the well-intentioned programmer and said to him “If you are going to chamfer using the 3D method, you had better push hard on the accelerator.” Not only was he running at a slow feed rate for this type of operation, but also he had programmed the step over moves as if he were machining a mold cavity that would need no polishing. He’s learning and probably won’t make that mistake again.

Because I am writing about this subject, I decided to pull up the program and check the feeds, speeds, and step over moves he was using to see how long the chamfering operation took at those settings.

The cutting parameters he was using were as follows: 7000 RPM, 100 in/min feed rate, and .002” step over, which collectively resulted in that operation taking twenty-nine minutes. He also chose to use a hog nose end mill with an .020R.

The biggest mistake I see with those settings is the small step over moves he was using, which is something you can’t adjust at the machine. The step over moves are set in the G-code.

I have the benefit of hindsight; it is easy for me to second guess his settings. But I probably would have chosen something closer to the following: 10,000

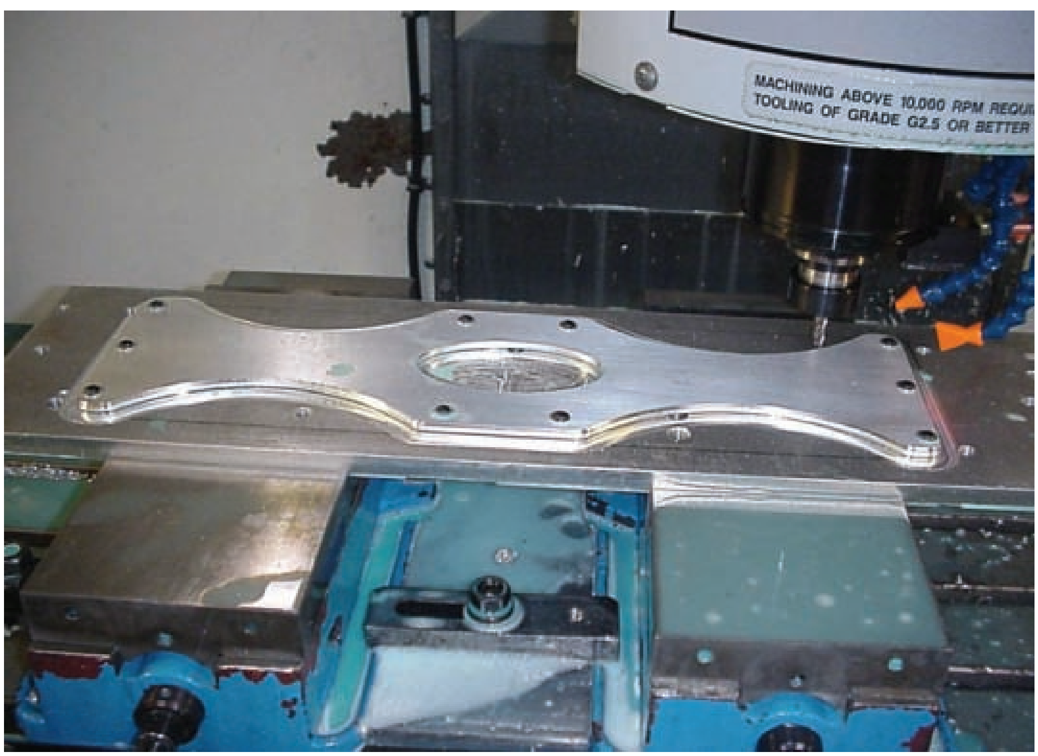


Figure 4-5 A chamfer is being 3D cut on the perimeter of this part.

RPM, 300 in/min feed rate, .006″ step over using a ball nose end mill. These setting would have resulted in the operation taking 3.8 minutes. If customers can't wait four minutes, then it's their problem.

5. Cut chamfers and edge radii using the 3D method. (see Fig. 4-6)

You can use a radius tool to cut radius edges; in the long run, it may be a bit faster. However, radius tools are a little tricky to setup and adjust.

Again, I prefer using either a ball nose end mill or hog nose end mill using the 3D cutting method to create radiused edges. That way the computer does all the thinking for you and little if any adjusting is necessary. Generally I use a ball nose end mill to cut soft material such as aluminum and a hog nose end mill to cut steels. A hog nose end mill will cut with less pressure than a ball nose because there is less cutting edge in contact with the material.

Cutting chamfers and edge radii are operations that need to be performed often now. I'll go so far as to say "too often." With fillets and chamfers being so easy to apply with CAD software, designers add them routinely to their solid part models just because they look "pretty." Many don't realize that chamfers and radii are much more time consuming to cut than they are to model.

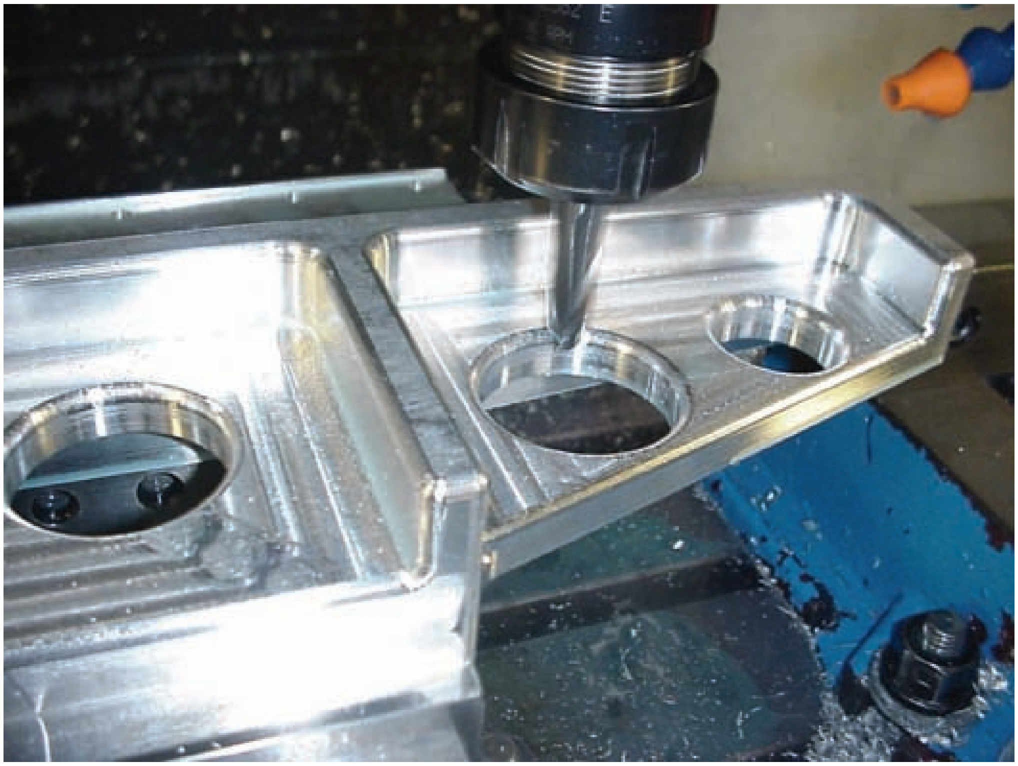


Figure 4-6 A ball nose end mill is shown cutting an edge radius around a hole using the 3D method.

Suggestions for Chamfering and Edge Dressing

1. Chamfering is not as easy as it looks.
2. For a chamfering tool to cut cleanly, the tip of the tool must be offset from the profile.
3. If you offset a chamfering tool to cut cleanly, the depth of the tool must be correct.
4. For most chamfering, use a ball nose or hog nose end mill, and program the chamfer using a 3D cut.
5. Cut chamfers and edge radii using the 3D method.

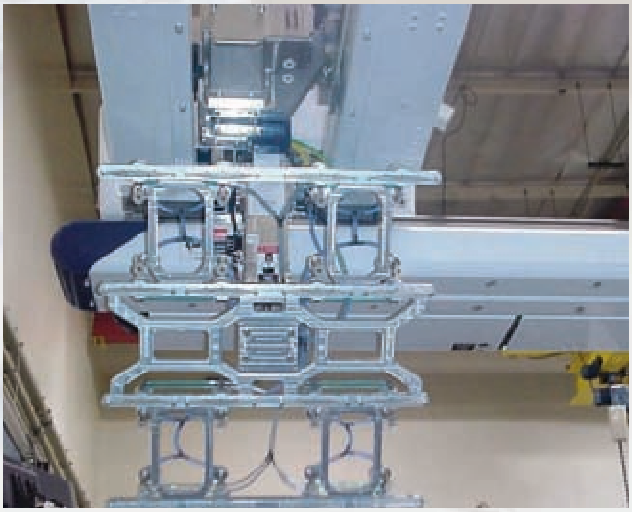
CHAPTER 5

More Help for Engineers

(from a machinist's point of view)

My experiences dealing with engineers over the years have been mainly positive. The majority of engineers I've worked with are solution seekers. They usually go out of their way to make things easier for people working on their projects.

However, as a result of the physical separation of engineering departments from shops, communication and feedback often suffer. This chapter is about providing feedback to engineers from a machinist's perspective. The goal is to help both groups work together more productively.



Engineers would do well to spend time talking with shop personnel to find out what obstacles machinists and fabricators face. Experienced mold makers, die makers, and machinists generally have good mechanical aptitude, are in tune with reality, and have worked on projects that didn't go so well, which they'd prefer to avoid.

If you bounce ideas off of them as you design your project, especially if you are unsure about something, you'll likely be ahead of the game.

1. It's a rare occurrence when a design works right off the drawing board. (see Fig. 5-1)

2. Catch mistakes at the computer.

Computers have been a blessing and a curse in manufacturing. A blessing because good designers have a tool that helps them design efficiently. A curse because careless designers have the same tool.

"Computer people" are accustomed to having things happen instantaneously. You can "fix" things on a computer quickly.

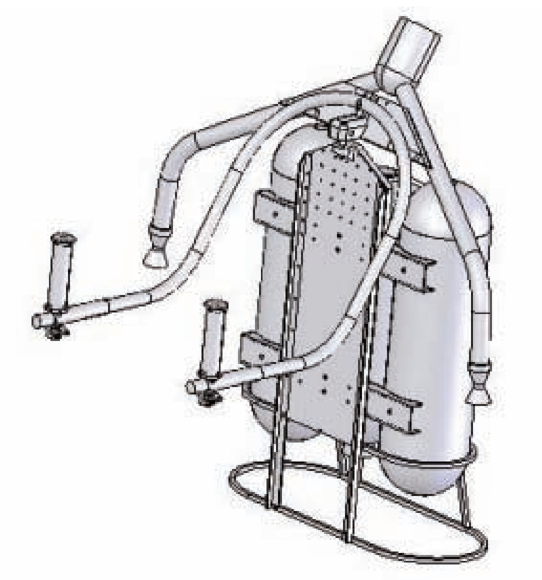


Figure 5-1 Rocket Pack designed by J. Harvey. Will it work in reality? Probably not.

What some people overlook is that the same spontaneity does not exist in manufacturing. Something as simple as a hole being off location can be a set back. Engineering and executing fixes are often expensive, stressful, and time consuming.

The best time to catch mistakes is at the computer, not when parts are being machined or assembled.

Take your time at the computer and put extra effort into getting things right the first time. Don't trust yourself on your first stab at designing or correcting something. The ripple effect of your work is often far reaching and will likely have a significant impact on cost and schedule.

3. Pay attention to details in your designs and drawings, and when you are working with shop personnel.

Machinists are detail-oriented people. They like things described clearly and accurately. Machinists often don't know the function of a part and rely on drawing to choose the tools and methods they use to machine parts. Getting accurate and clearly defined drawings to the shop floor is an engineer's job. The devil is in the details.

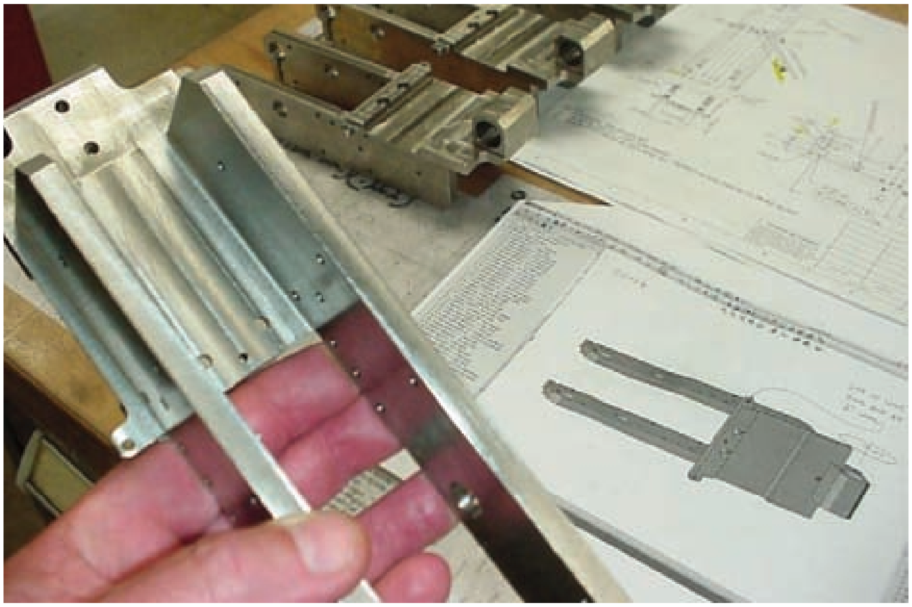


Figure 5-2 For shop personnel, the best information an engineer can provide is both a CAD model and detailed drawing with tolerances. Second best is a detailed drawing. With a drawing only, a machinist or programmer is forced to construct a computer model, which opens the doors for errors. Last, a computer model only is sometimes sufficient, but leaves tolerance interpretation open to the machinist.

4. Submit both a CAD model and detailed drawings for smooth sailing. (see Fig. 5-2)

Occasionally an engineer will submit only a CAD model to save time. If a production line is down or there is some other emergency, submitting a CAD model only can save time, but comes at the expense of leaving tolerance interpretation open to the machinist or programmer.

Experienced machinists can generally tell from a CAD model what dimensions are important…but not always. For example, a machinist wouldn't know the difference between precisely located dowel pin holes and drilled holes just from a model. As an engineer, the best information you can provide to a shop is both a CAD model and a detailed drawing with tolerances.

5. Avoid initiating a poorly planned project for machining and fabricating just because you or somebody else wants to get a project "moving."

New projects have enough unpredictability built into them already. If you've been in manufacturing for any length of time you've probably been involved with projects that fought you from the moment of conception to the day they died.

Getting off to a good start is critical to getting the upper hand on a project and keeping it on track. Good designing, planning and drawings start at the engineering level. When machinists and fabricators take over, it is equally important that they get off to a good start with well thought out planning and execution. Hiccups along the way are to be expected. But if the project starts off on a solid foundation, the hiccups are easier to deal with.

6. Avoid cramming too much information into one drawing. (see Fig. 5-3)

We often see drawings where a mirror image of the same part is called for. Generally that doesn't cause confusion. However, when mirror images are called for with exceptions, that's when things can go south in a hurry.

A recent example of this confusion and the resulting loss of time and money happened with a blow mold project I was involved with. The molds are used in pairs in the molding machine—one right hand and one left hand. Each mold has a front half and a back half. In addition to the molds and mold bases being mirror images, the cooling lines for the front half and the back half of the mold bases are a little different. The mounting holes on the front half and the back half are also a little different. This was all described in one drawing.

To make a long story short, when the mold bases came back from the vendor they looked like Swiss cheese and couldn't be saved. I begged the engineer for the next attempt to make four separate drawings, one for each mold base. That didn't seem to make sense to him, but it made sense to me knowing that vendors just want to get jobs done as quickly as possible.

One of your goals as an engineer should be to provide information that lets the machinists and fabricators do as little thinking as possible. Ideally, fabricators should spend their time fabricating, not spend their time interpreting confusing drawings and instructions.

The engineer made the four separate drawings which took him maybe a couple of hours. We now have four good mold bases, albeit two months late and about $20,000 over budget. Paper is cheap. Use it to avoid confusion.

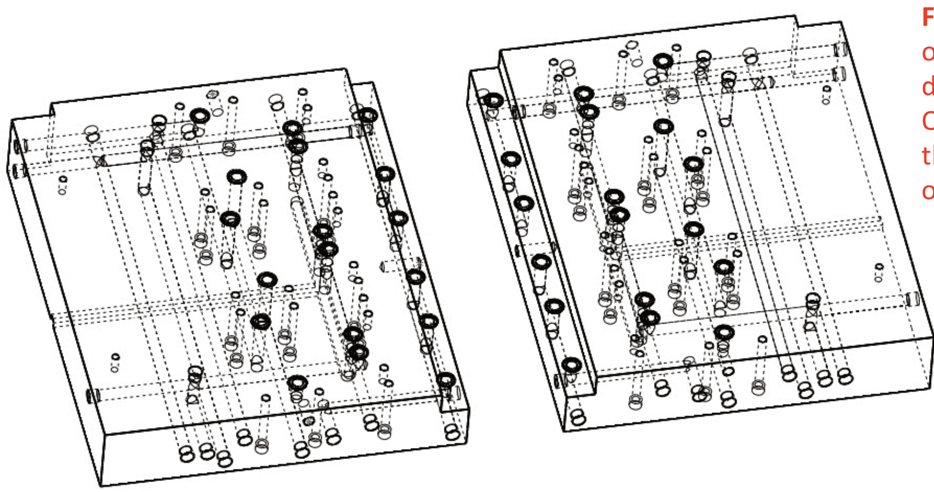


Figure 5-3 Mirror images of parts made from one drawing are often needed. One tip I can give you is that mirror images always open like a book.

7. Avoid adding fillets, radii, and chamfers flippantly on parts just because it is easy to do with a computer.

Fillets, chamfers, and radii can substantially increase the time it takes to make parts. Special cutters may be needed; additional programming may be needed, etc. So if you are concerned with economy at all, try to keep things simple. Stick with some of the old-fashioned edge dressing callouts such as "Break all sharp edges .02."

8. Avoid showing a flat bottom hole in drawings unless you want the hole to be flat bottomed.

This can be confusing to a machinist, especially if it is not obvious to the machinist how the hole will be used. Showing a drill tip angle at the bottom of a hole will remove any doubt as to what you want and will be faster for the machinist to produce. Flat bottom holes require an extra tool or operation to create the flat bottom.

9.Be aware that milled pockets have to have a radius somewhere. (see Fig. 5-4)

If you submit a drawing of a part that has pockets or windows with square corners, the job may get sent out for wire cutting or electro-discharge machining (EDM) at a substantial cost increase or delay to your project. This is a common oversight we see.

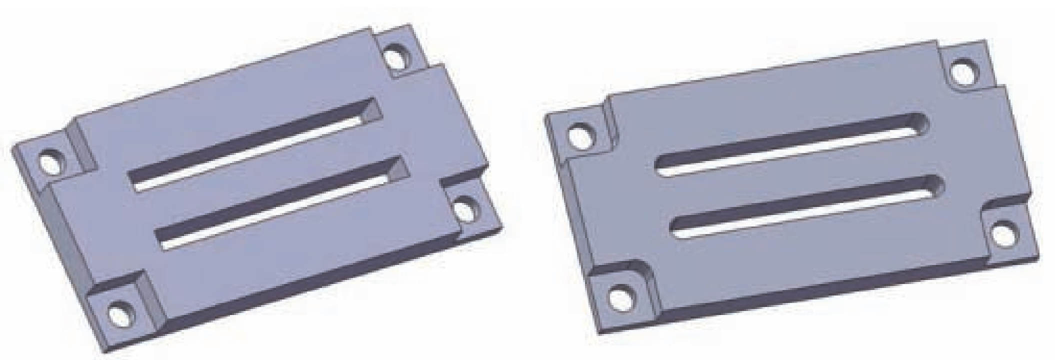


Figure 5-4 Square corners are nearly impossible to cut with a milling machine. Include radii on pockets and cut outs when possible.

10. Make a cardboard template of brackets added to a machine after the fact. (see Fig 5-5)

Brackets are one of those things that look easy—but that's exactly why they get screwed up. Simple jobs get screwed up a lot, which is why I like to adhere to the principal that "there's no such thing as a simple job." Often a bracket is needed to add a sensor or limit switch to a machine after it's been built.

When designing and creating a drawing for a bracket, avoid holding up a ruler to a machine and saying, "That looks about right." I can almost guarantee you that approach won't work.

Instead, create a cardboard template of the bracket you are designing before submitting a drawing to the shop for fabrication. Go to the machine where the bracket is going to be used, cut the cardboard with a pair of scissors, and bend it until the cardboard model fits correctly. Mark the hole locations on the cardboard. Take your time. Then make a drawing of the bracket using the cardboard model as a reference.

Recently we had to re-make a simple bracket a dozen times until it fit right on a machine. I have little patience for that kind of careless engineering. Fortunately I wasn't the one working with the engineer. Some machinists see this type of inefficiency as job security. Vendors may love it because they can charge for each revision. I see it as wasting my time.

As one of my respected mentors used to say: "It's easy to make junk."

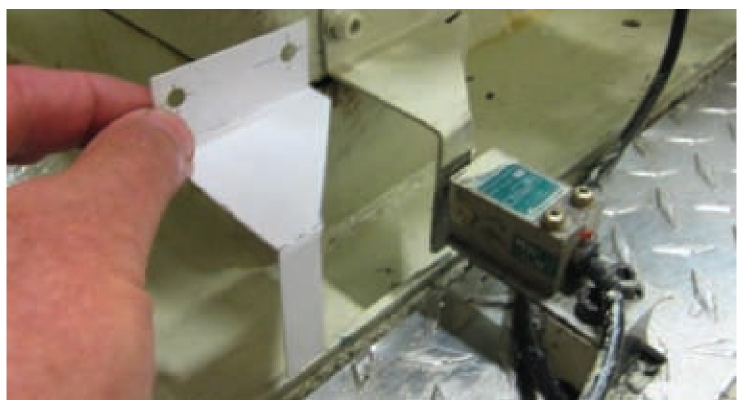


Figure 5-5 Low-tech still has its place. It's a good idea to construct a cardboard model of an odd-shaped bracket to see how it fits on a machine before releasing a drawing. You'll likely end up with a bracket that works the first time around.

Figure 5-6 Small inside corner radii take longer to machine. Did this window really need the inside corner radii that small? I doubt it.

11. Be aware that small inside radii take longer to machine. (see Fig 5-6)

For machinists, inside radii are more of a concern than outside radii. The larger you can make inside radii in your part designs, the more options you give the machinist in terms of choosing end mill sizes to cut the radius. The larger the end mill, the faster the part can be made.

12. Try to maintain consistency in the standard of hardware you use.

In other words, stay with either metric or English hardware in subsequent add-ons or modications to a machine or tool. If a machine was built with metric hardware, then continue design modifications using that standard.

13. If possible, try to design parts with parallel surfaces somewhere. (see Figs. 5-7and 5-8)

Parallel surfaces are used by machinists to hold parts. If no such surfaces exist, then the part becomes difficult to make. Figure 5-7 shows a seemingly simple part that had me puzzled. There was no easy way to hold the part for machining.

I decided to hold the part for machining by adding an extension to the part, as shown in Figure 5-8. After machining the part, I cut the extension off and disk sanded the remaining nub flush with the face. Could the part have been designed with parallel surfaces somewhere so that it would have been easier to make? The answer is probably yes.

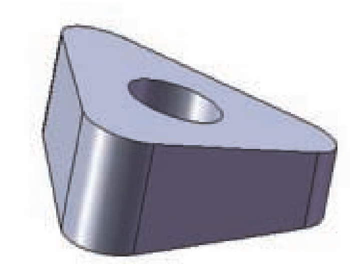


Figure 5-7 This part has no parallel surfaces, which makes the part difficult to hold for machining. When possible, design parts with parallel surfaces.

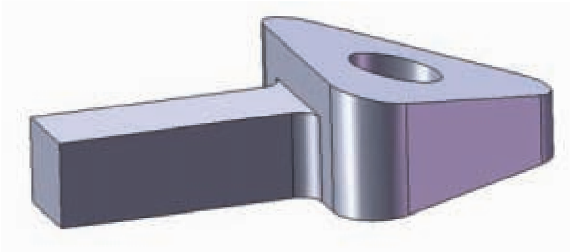


Figure 5-8 An extension had to be added to this part to hold the part for machining.

14. Call out chamfers correctly. (see Fig. 5-9)

I had an issue with this recently. I was adding a chamfer to an existing part and the chamfer was called out on the drawing as shown on the left:

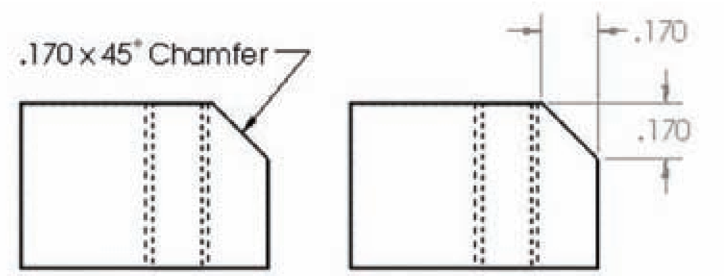


Figure 5-9 Chamfer dimensions, with leader lines as shown on the left, leave doubt as to what the designer really wants. To avoid confusion, it is better to dimension chamfers as shown on the right.

I touched off the 45-degree cutter to the corner of the part and proceeded to cut in .170" as I normally would. I noticed the chamfer was starting to come close to the edge of the tapped hole and hadn't moved the cutter anywhere close to .170". That's when I determined the designer had not described correctly what he wanted. The callout shown does not mean the face of the chamfer is .170" in length. It means the adjacent sides of the "triangle" are .170" in length.

15. Pay attention to angle callouts for countersunk screw holes.

This is another common oversight. Like it or not, we now live in an English/metric world. Flat head screws that are tightened down into countersunk holes with angles that don't match will not hold well. They will not provide secure and stable fastening.

The included angle of a metric countersunk hole is 90 degrees. Standard English countersunk holes are 82 degrees. Attention to this detail is important.

16. Help shop personnel by ordering special hardware needed for your projects.

If there are special materials or hardware needed for your project, you should order them. When a customer brings material and hardware to the shop, it is appreciated. It vastly reduces middlemen, ordering issues, availability issues, communication issues, cost issues, etc. Furthermore, the shop will likely get started on your project sooner rather than later.

Suggestions for More Help for Engineers

1. It's a rare occurrence when a design works right off the drawing board.
2. Catch mistakes at the computer.
3. Pay attention to details in your designs and drawings, and when you work with shop personnel.
4. Submit both a CAD model and detailed drawings for smooth sailing.
5. Avoid initiating a poorly planned project for machining and fabrication just because you or somebody else wants to get a project "moving."
6. Avoid cramming too much information into one drawing.
7. Avoid adding fillets, radii, and chamfers flippantly on parts just because it is easy to do with a computer.
8. Avoid showing a flat bottom hole in drawings unless you want the hole to be flat bottomed.
9. Be aware that milled pockets have to have a radius somewhere.
10. Make a cardboard template of brackets added to a machine after the fact.
11. Be aware that the small inside radii take longer to machine.
12. Try to maintain consistency in the standard of hardware you use.
13. If possible, try to design parts with parallel surfaces somewhere.
14. Call out chamfers correctly.
15. Pay attention to angle callouts for countersunk screw holes.
16. Help shop personnel by ordering special hardware needed for your projects.

CHAPTER 6

Becoming Familiar with CAD

CNC machining takes a blend of computer skills and machining skills. Some competent machinists, especially the old timers, lack the computer skills needed to go from print to part using CNC machines. I lacked the computer skills at one time, but couldn't tolerate not having complete control of jobs I was running. Therefore, I taught myself the computer side of CNC machining so I could create my own programs. I learned mostly through trial and error, which is not the greatest way to learn, but so be it.

My approach to learning software was inspired by my twelve-year-old daughter. That was about twenty years ago.

Because I had little prior experience, I was reluctant to have anything to do with computers. My daughter was already having great fun with the computer and was doing what seemed like complex tasks with it.

1. Often the best way to start is simply to start.

One day I asked my daughter if she would teach me how to do word processing and she said "Sure." She instructed me to click on the Microsoft Word icon, which I did. I had a sinking feeling in the pit of my stomach. I was envisioning having to deal with a myriad of complex menus, icons, and computer lingo that would take me the next decade or so to master.

When the program launched and the default page came up, I reluctantly asked her, "So what do I do now, Joanna?" Her answer just about floored me and I'll never forget it. She said, "Just start typing."

That day taught me two things about computers. First, you have to have an interest in learning. Second, sometimes you have to "just do it."

2. Paying attention to detail and working methodically often leads to positive results. (see Fig. 6-1)

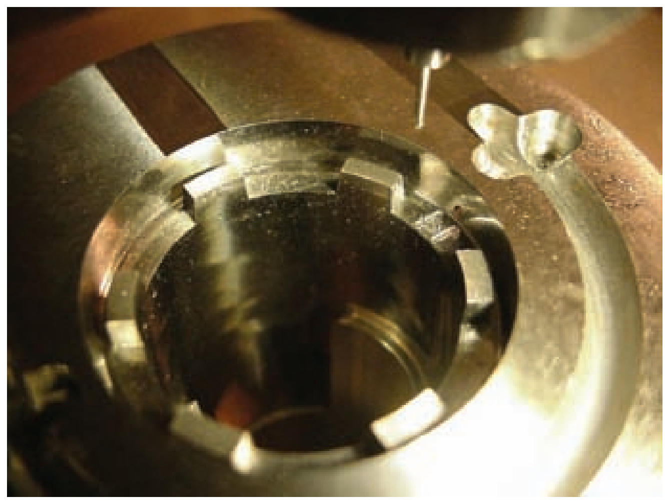


Figure 6-1 Caution is advised when engraving expensive mold cavities.

Manipulating a computer is probably my weakest area as it relates to CNC machining. I'm not as nimble with a computer as many are from the younger generation. I may have started too late in life.

Even if you can't manipulate a computer with blinding speed, that may actually work in your favor in this line of work. You'll compensate by focusing on details and by methodically checking your work.

My daughter, by the way, can't stand watching me slog along with a computer. I believe it's a generational thing. With all her innate ability to fly along with a computer, she may be the last person I'd trust to engrave expensive mold cavities. I'd rather do that job myself!

3. Stick with an application that is comfortable for you.

I'm not too swift at learning software. Therefore, once I do learn an application, I tend to stick with it. I'm reasonably comfortable using our current CAD/CAM system, which is SolidWorks® on the modeling side and Camworks® on the machining side. I now prefer doing all phases of a project from modeling parts on the computer to planning, programming, and machining. Taking ownership of a project vastly cuts down on communication errors and rework. When I can, I like being involved with designs as well.

4. People have different ways they like to learn.

Some people prefer going to school, some people prefer going through tutorials. I started by going through a simple tutorial in SolidWorks. I was fortunate to have a seasoned SolidWorks® mentor to fall back on when I got stuck. When I completed the first tutorial and modeled a simple part, the mentor said to me "Now you know you can do it. From here on out it's endless." He was right; as with many industrial computer applications, the rabbit hole runs deep. Not only is it endless, but it's also constantly changing as new versions of software come out.

Because I'm primarily a machinist who needs to get things done, I tend not to stray too far off the beaten path with unorthodox ways of programming and machining. I try to keep things simple, focusing on what I need to know to get a good product in a reasonable amount of time.

After completing the first SolidWorks tutorial, I was off and running—and then switched to my preferred method of learning, which is having a project to accomplish. The project doesn't have to be complicated, just useful in some way. By having a project to accomplish, you immediately start to weed out the necessary from the unnecessary.

This "project" approach has downsides. It is easy to develop bad habits. I could have avoided many bad habits if I had the patience to take classes or go through more tutorials.

Next, we're going to provide an overview of how to model an adapter plate (see. Fig. 6-2). We'll use SolidWorks; it's popular, user-friendly,—and what I use. You can download a student version from the Internet.
The purpose here is to show you in a fundamental way how parts are made using CAD/CAM technology. Once you can model this adapter plate, you'll know many of the basic concepts needed for modeling. After that…it's endless!

For those who are already experienced at using SolidWorks, I'll present a few seemingly simple models at the end; they can be somewhat perplexing to construct. At least they were for me!

Figure 6-2 This is the finished adapter plate we'll be modeling, programming, and machining. It's a simple part, but once you can make one of these using CAD/CAM equipment, you'll have a good foundation for producing more complex parts.

5. The first step is to create a computer solid model. (see Fig. 6-3)

One of the beauties of SolidWorks is that you don't have to be a guru to put it to good use in a machine shop.

One hurdle we face in our shop is interpreting lousy hand sketches submitted by maintenance mechanics and other customers. At my suggestion, some of the more ambitious customers have learned to model parts using SolidWorks.

This approach makes my job as a machinist easier. Once someone provides me with a SolidWorks model, I know two things right away. First, they are happy with what they've modeled, which will likely cut down on rework. Second, I'll have all the dimensions I need to make the part.

The downside is that we now have maintenance mechanics "gone wild." Some are now modeling and inventing all kinds of fancy wrenches, brackets, and other parts.

One guy came in with a wrench he modeled that had tapered and lofted surfaces. He filleted and radiused nearly everything. Apparently he thought—like many people do—that the only thing a machinist has to do is press a few buttons and out pops the part. (That's not true, by the way!)

I directed him to get rid of that ridiculous model and then re-model the wrench using a simple 2D profile. For the angle, I suggested he clamp the wrench in a vise, then whack it with a hammer a few times. Afterward, we heat-treated the wrench. We now have a happy customer, and one who will be leery of over-engineering anything for a while.

A reminder: The goal in this chapter is to provide an overview of the modeling process, not the details, for which there are better sources. One such comprehensive book is *SolidWorks for Technology and Engineering* by James Valentino and Nicholas DiZinno.

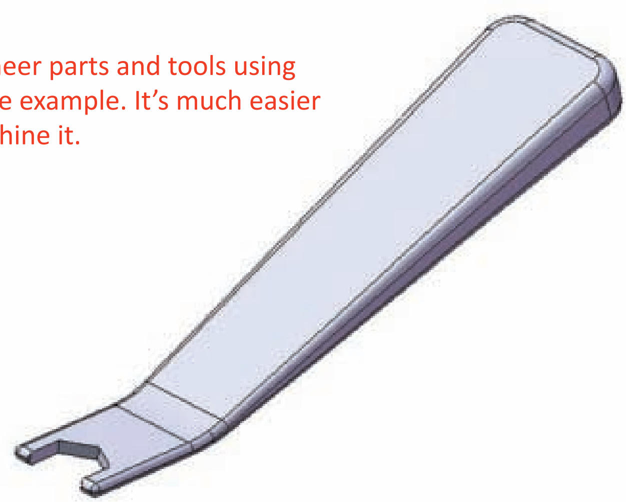


Figure 6-3 Designers often over-engineer parts and tools using SolidWorks. This silly wrench is a prime example. It's much easier to model this wrench than it is to machine it.

SolidWorks tutorials that come with the software are also helpful if you want to become a better modeler. Tutorials take you through example parts step by step. Each model in a tutorial explores a different SolidWorks capability. In this book, you'll be exposed to a visual progression of the fundamental steps taken to model an adapter plate. You'll notice the modeling progression for the adapter plate is quite straightforward and incorporates commonly used SolidWorks capabilities.

Let's proceed: the adapter plate drawing is shown in Figure 6-4.

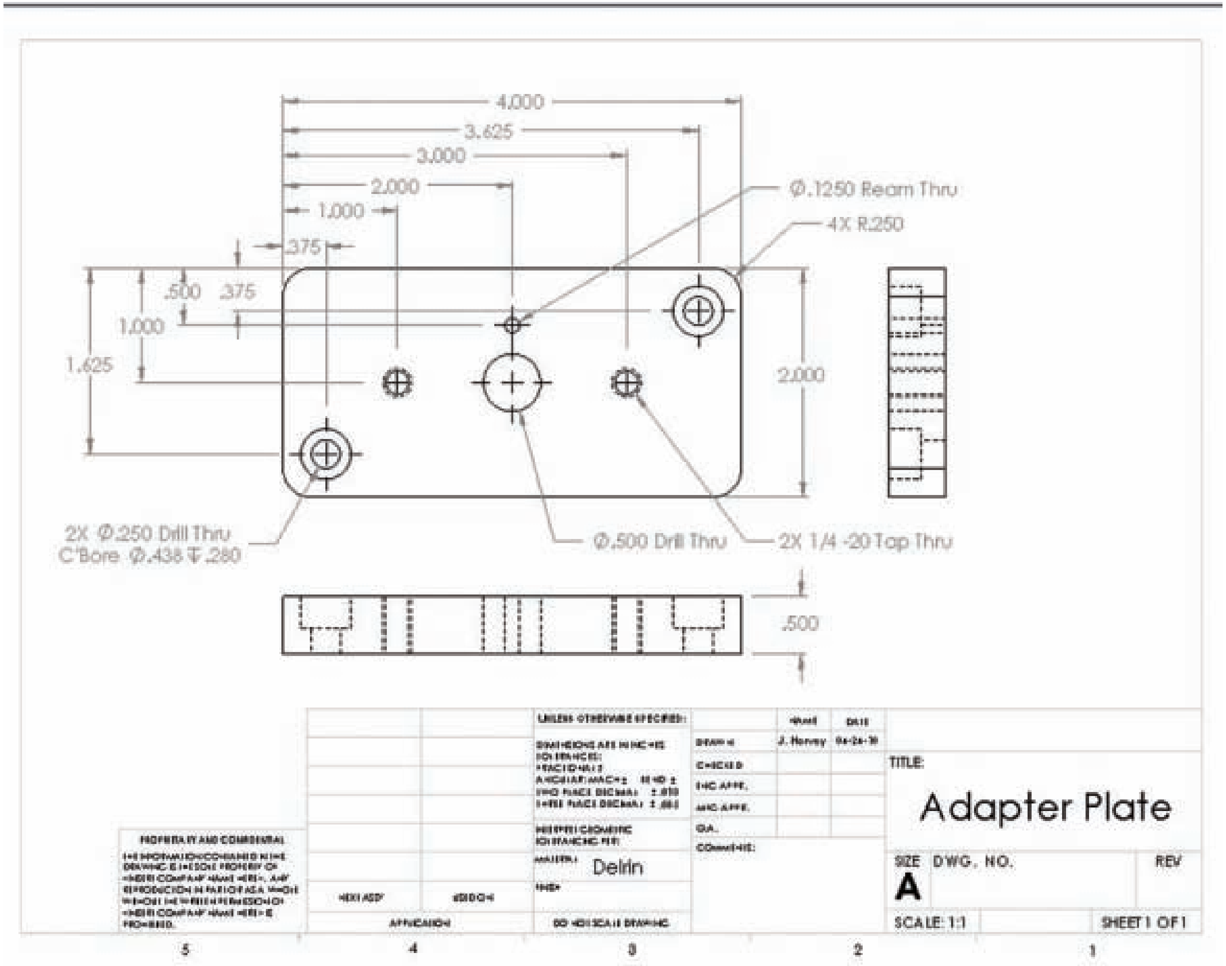


Figure 6-4 Three-view drawing of adapter plate.

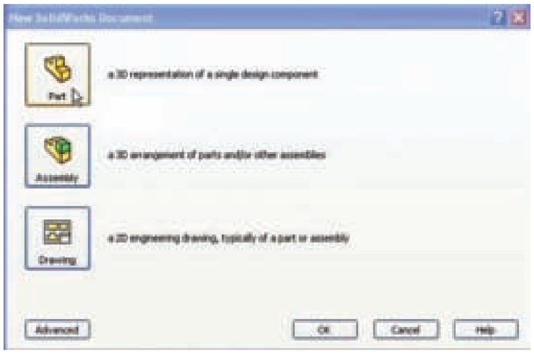


Figure 6-5a Select what you want to create. In this case, we want to model a "part" — the adapter plate.

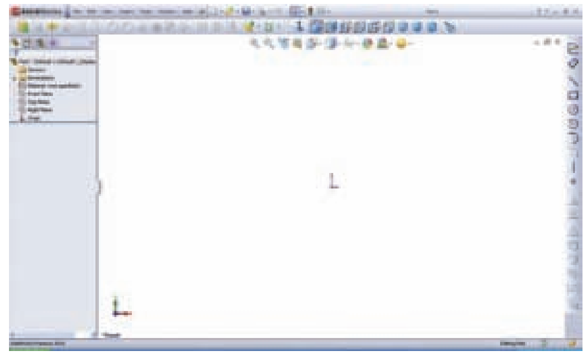


Figure 6-5b Main modeling screen with only the origin showing.

6. Launch the main modeling screen. (see Figs. 6-5 and 6-6)

How you set up the SolidWorks toolbars is a matter of personal preference. The screenshot in Figure 6-5b is what I learned on. It is simple, but practical for what I do—I model parts for machining. Today we use a more comprehensive SolidWorks default screen, the "Command Manager," shown in Figure 6-6. It provides numerous icons on a single screen for manipulating and analyzing models.

For simplification, I've chosen to model the adapter plate in the following examples, using the screen I used when I first learned.

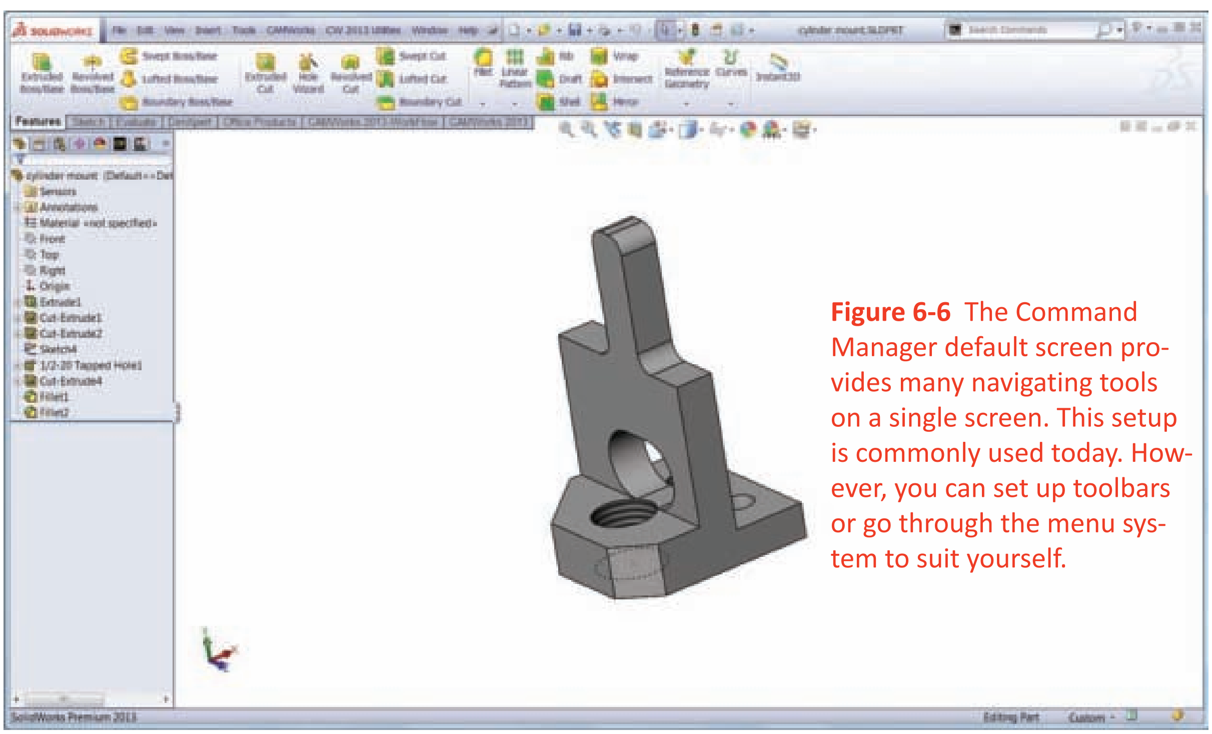


Figure 6-6 The Command Manager default screen provides many navigating tools on a single screen. This setup is commonly used today. However, you can set up toolbars or go through the menu system to suit yourself.

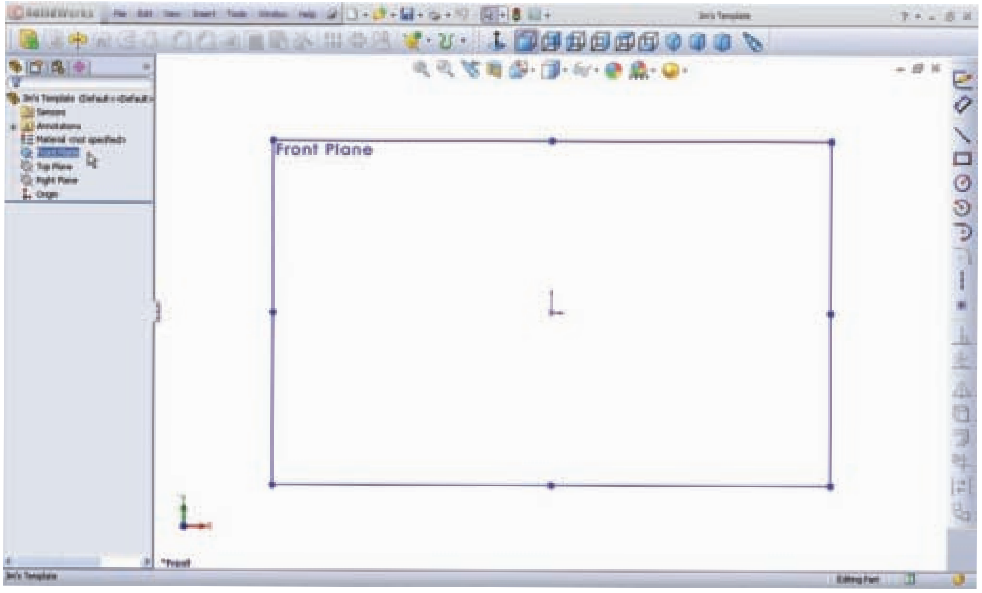


Figure 6-7 Select a plane to begin modeling. In this case, the front plane was chosen.

7. Begin modeling by choosing a plane. (see Fig. 6-7)

You can choose any plane to begin modeling. As a matter of consistency, though, it is common to choose the front plane to begin. The front plane is the first plane shown in the features tree and can be chosen by clicking the words "Front Plane" in the features tree.

On the front plane, begin sketching the drawing view that shows the most surface area and dimensioned features. This view commonly includes the overall length and width of a part and many of the holes in the part. The adapter plate example will be modeled this way.

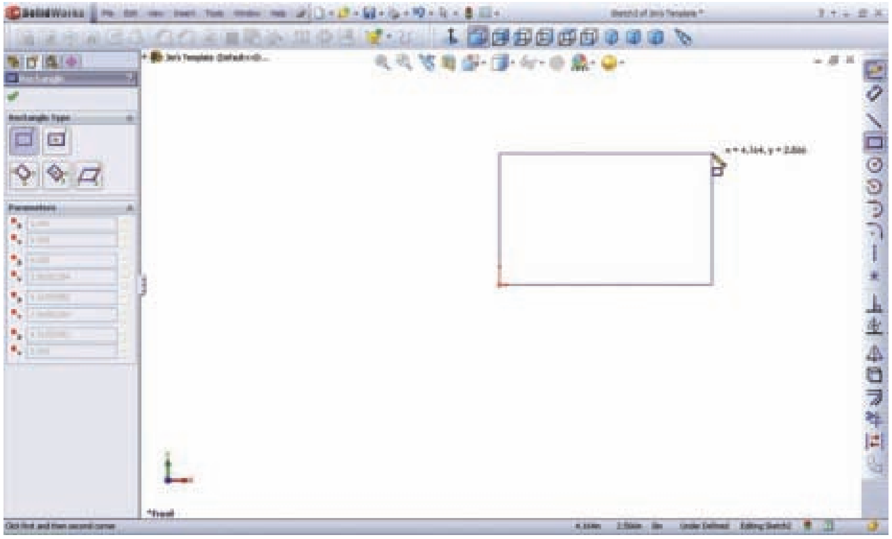


Figure 6-8 Starting at the origin, sketch a rectangle approximately the size of the adapter plate.

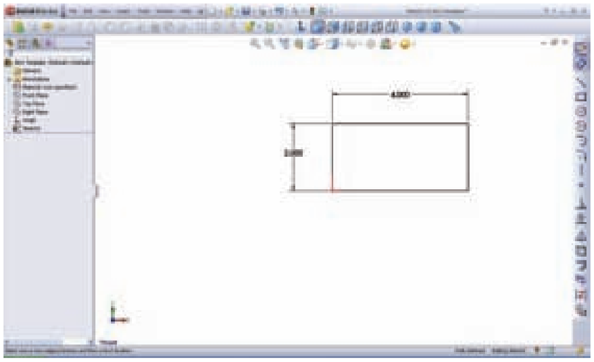


Figure 6-9 Dimension the rectangle to 4" wide x 2" high.

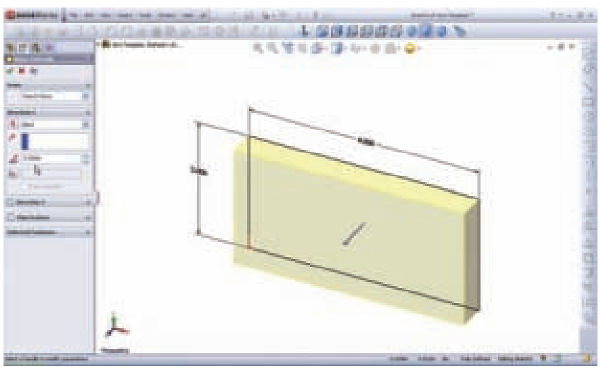


Figure 6-10 Extrude the rectangle to a 1/2" depth.

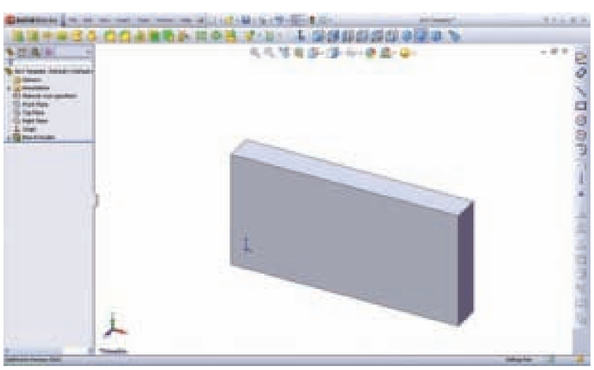


Figure 6-11 Extruded rectangle completed.

8. Develop the basic shape of the adapter plate. (see Fig. 6-8)

Open a sketch. Then starting at the origin, draw a rectangle that roughly represents the perimeter of the adapter plate.

9. Dimension and extrude the rectangle that represents the adapter plate. (see Figs. 6-9, 6-10, and 6-11)

Extruding a two-dimensional sketch into a three-dimensional model is a simple yet powerful tool for constructing CAD models. SolidWorks makes this easy. Any two dimensional sketch on any plane can be exruded or thrust out to become a three-dimensional shape. For this adapter plate, the overall length and width of the adapter plate is extruded to 1/2" using the Boss Extrude icon

Conversely, the Cut Extrude icon cuts away material from a model using any two-dimensional sketch created by the designer. Likely, the two most used icons in SolidWorks are Boss Extrude d Cut Extrude .

When extruding the first sketch, which in this case is the overall length and width, SolidWorks automatically puts the model into an isometric view, as shown in Figure 6-10.

10. Once the overall shape of the adapter plate is complete, focus on creating the holes. (see Figs. 6-12 through 6-19)

To create the holes, we first sketch them generally in the model (see Fig. 6-12), then set the dimension of each hole (see Fig. 6-13). We then specify the location of each hole within the adapter plate (see Fig. 6-14).

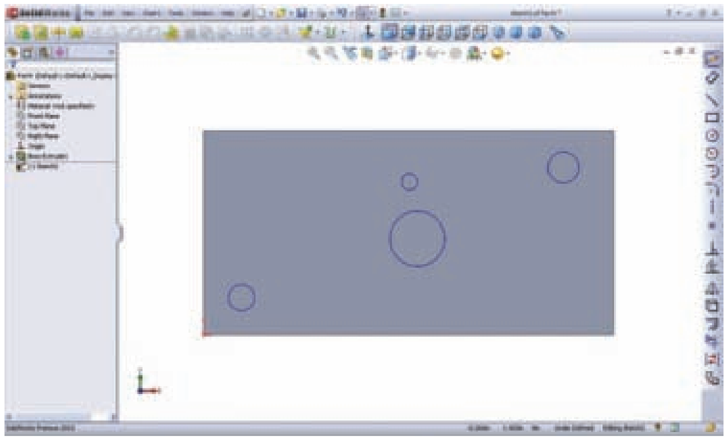


Figure 6-12 Begin creating holes by sketching circles on the plate.

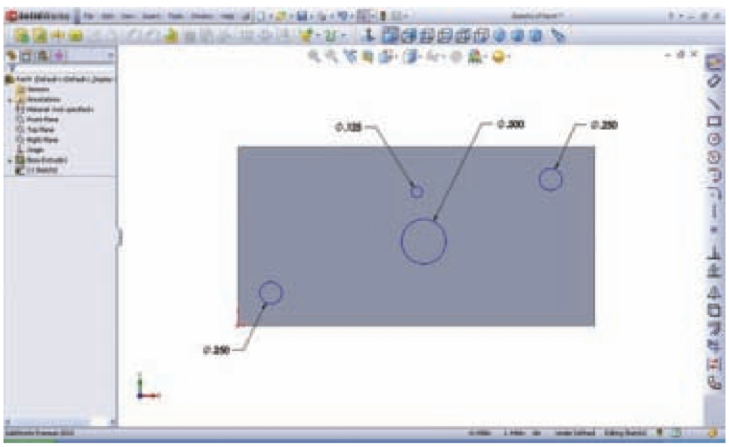


Figure 6-13 Dimension the diameters of the circles.

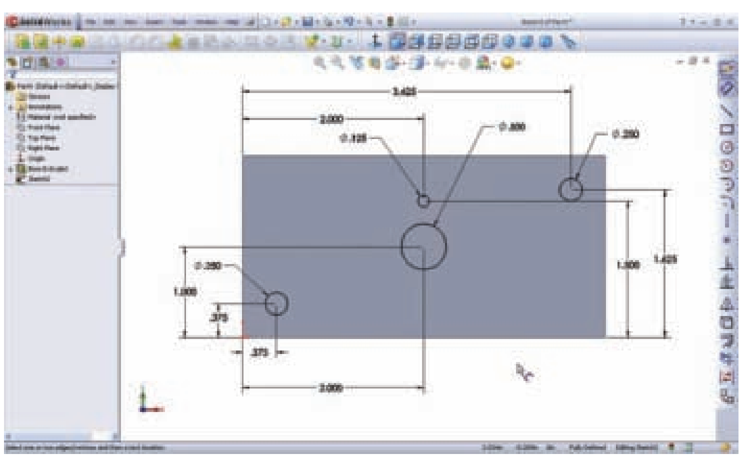


Figure 6-14 Dimension the locations of the circles.

After the circles are dimensioned and located, cut extrude the circles to create finished holes in the model. Do this by clicking the Cut Extrude icon,  selecting “through all” in the dialogue box, and clicking “OK” (see. Figs. 6-15 and 6-16). Circles can now be sketched and extruded to develop the counter bored holes for the 1/4–20 bolts (see Fig. 6-17).

As a side note, I usually prefer to sketch and extrude counter bored holes rather than use the Hole Wizard,  which provides default hole sizes for a given screw. I prefer to sketch and extrude counter bored holes because it gives me the option either to quickly change the diameter of a counter bore to match an end mill size I want to use or to change the depth of the counter bore if the part is a little thin.

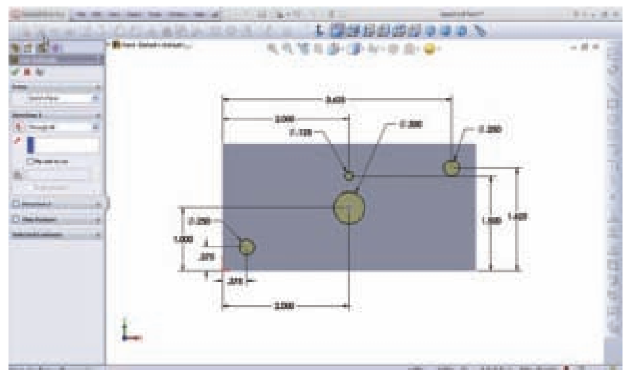


Figure 6-15 Cut holes in the model by cut extruding the dimensioned circles through.

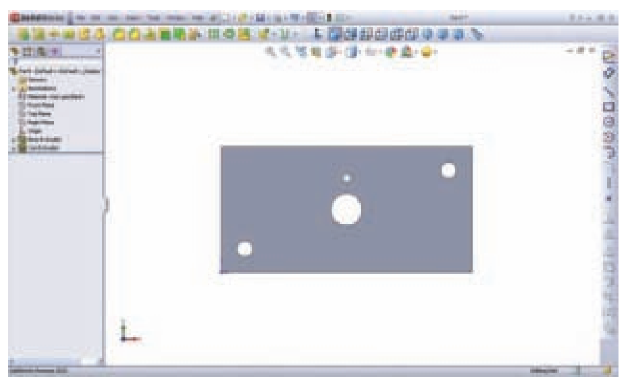


Figure 6-16 Extruded holes completed.

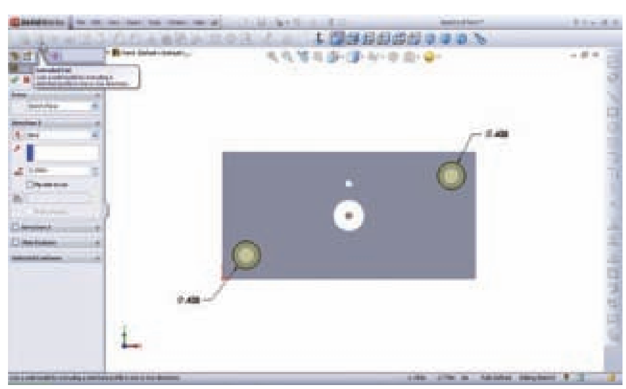


Figure 6-17 Sketch the counter bored holes and cut extrude them to the proper depth.

Next, install the tapped holes using the Hole Wizard program and set the dimensions for their locations (see Figs. 6-18 and 6-19). For this adapter plate, choose a 1/4–20 straight tap and select "through all" for the end condition.

SolidWorks includes a support utility, the Hole Wizard , that lets you install standard size holes of various configurations in a model. These standard hole types include counter bored holes, counter sunk holes, straight holes with a drill tip angle, straight tapped holes, and pipe tapped holes.

The beauty of using the Hole Wizard is that you now have a record of the type of hole installed, which is shown in the Features Tree. I find this record especially useful with tapped holes — when it comes to programming the part for machining, I can look back at the Features Tree and know exactly what tap to use.

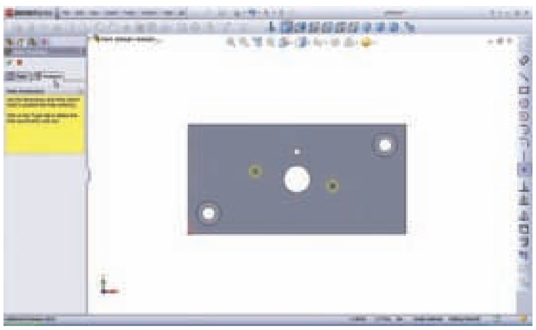


Figure 6-18 Create the 1/4-20 tapped holes in the model using the Hole Wizard.

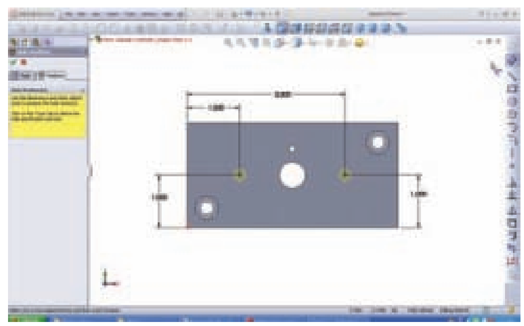


Figure 6-19 Dimension the locations of the ¼-20 tapped holes.

11. Apply the corner radii to complete the model. (see Figs. 6-20 and 6-21)

Once you've completed this step, that's it! The model is done and ready for programming.

Although this is a relatively simple model, many of the fundamental concepts of SolidWorks were used. From here on out, it's endless. Again, one of the beauties of SolidWorks is that you don't have to be a guru to put it to good use in a machine shop.

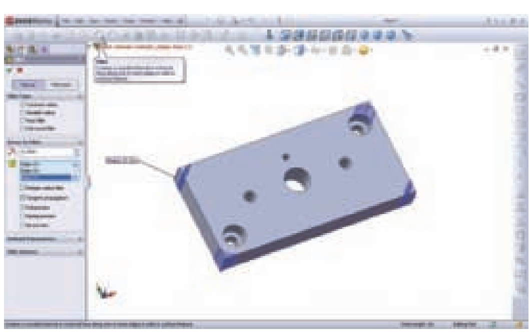


Figure 6-20 Apply the corner radii.

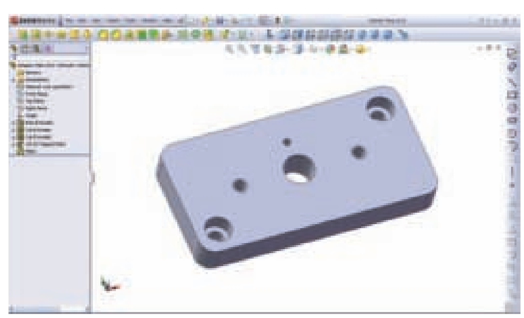


Figure 6-21 Completed model.

Many jobs that come through a machine shop are no more complicated than this. In turn, modeling these parts is no more complicated than what has been shown thus far. I've made hundreds of different parts using the simple strategy outlined in this chapter.

12. Practice constructing a few perplexing solid models. (see Figs. 6-22 through 6-26)

As promised, I've included a few relatively perplexing models for people who are already good at using SolidWorks. The word *perplexing* is used loosely here because the following models may not, in fact, be perplexing for some people. They were for me!

If you are unsuccessful constructing these models, try to find a SolidWorks guru who is willing to help. Sit down next to the person and watch the strategy. The person will likely discard a few initial strategies, but that's part of the modeling process. There is no right or wrong way to construct a model, but some strategies work better than others. Another way to get assistance is to visit a SolidWorks discussion group on the Internet.

a) Three-sided pyramid (see Fig. 6-22)

The first solid model presented is a three-sided pyramid (plus the base). The challenge here is that all six seams of the pyramid should be equal in length. The lengths of the seams are irrelevant. However, to give you a starting point, I made the seams three inches in length. To make this challenge more interesting, do not use 3D sketch mode.

b) Gemstone (see Fig. 6-23)

The second model is a gemstone of some sort. I modeled the gemstone using a large fake gemstone as a guide. Use whatever angles, dimensions, and number of facets you think looks best.

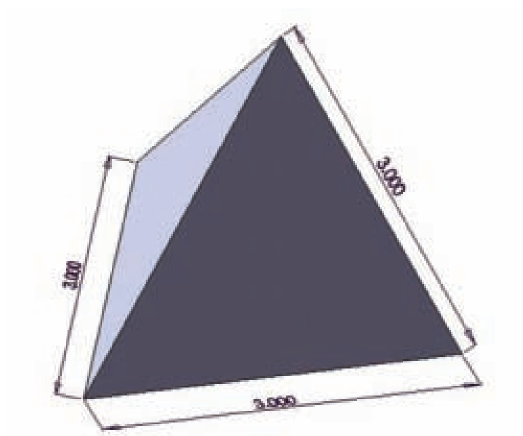


Figure 6-22 Three-sided pyramid.

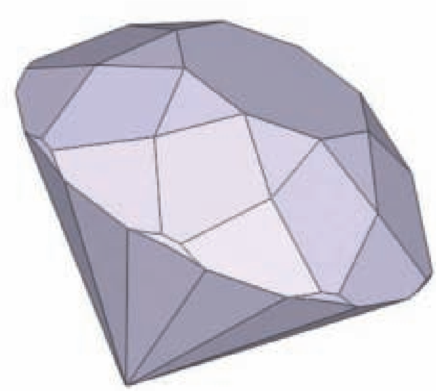


Figure 6-23 Gemstone.

c) Spade drill (see Fig. 6-24)

The third model is a spade drill. The challenge here is that the top cutting edges of the drill should extend to the outside "diameter." Use whatever angles you think will make your model look similar to the one shown. Carbide spade drills like this work well for cutting hard material. The hex shape of the drill provides chip clearance so they won't gall as easily as round spade drills.

d) Bolt (see Fig. 6-25)

The fourth model is a bolt. The bolt size is irrelevant, but to give you a starting point, I modeled a 1/4–20 bolt. The challenge with this one is that the last thread, near the shoulder of the bolt, should blend to the outside diameter in one turn.

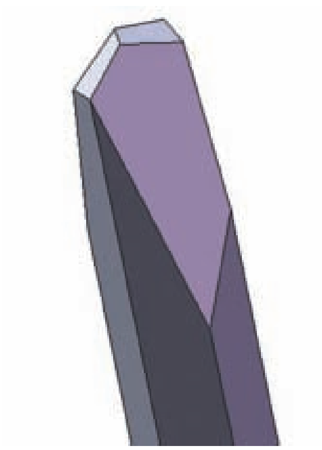


Figure 6-24 Spade drill.

Figure 6-25 ¼-20 bolt.

13. Assemblies can be useful when designing indexing fixtures for machining parts. (see Fig. 12-26).

Otherwise, in the context of machining parts, SolidWorks assemblies are a bit outside the scope of what most machinists work with.

Constructing assemblies in SolidWorks is relatively straightforward. However, if you are interested in learning how to construct them, I suggest you work through some tutorials or find someone who is willing to help. You'll quickly catch on to the basics of constructing simple assemblies. After that, of course, it's endless!

This concludes this introduction in using SolidWorks. It is important to learn modeling to get to our ultimate goal, which is to machine parts using CNC equipment.

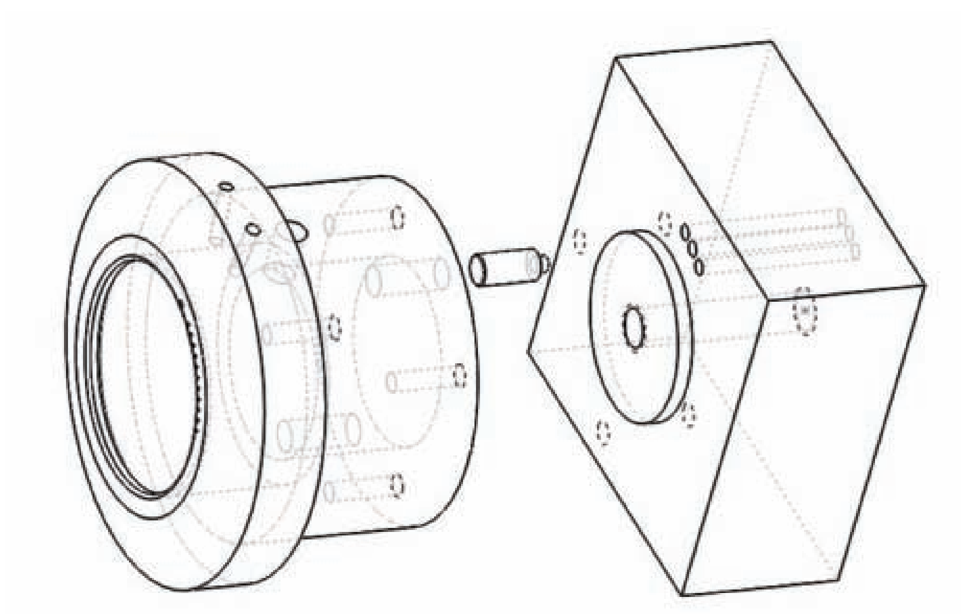
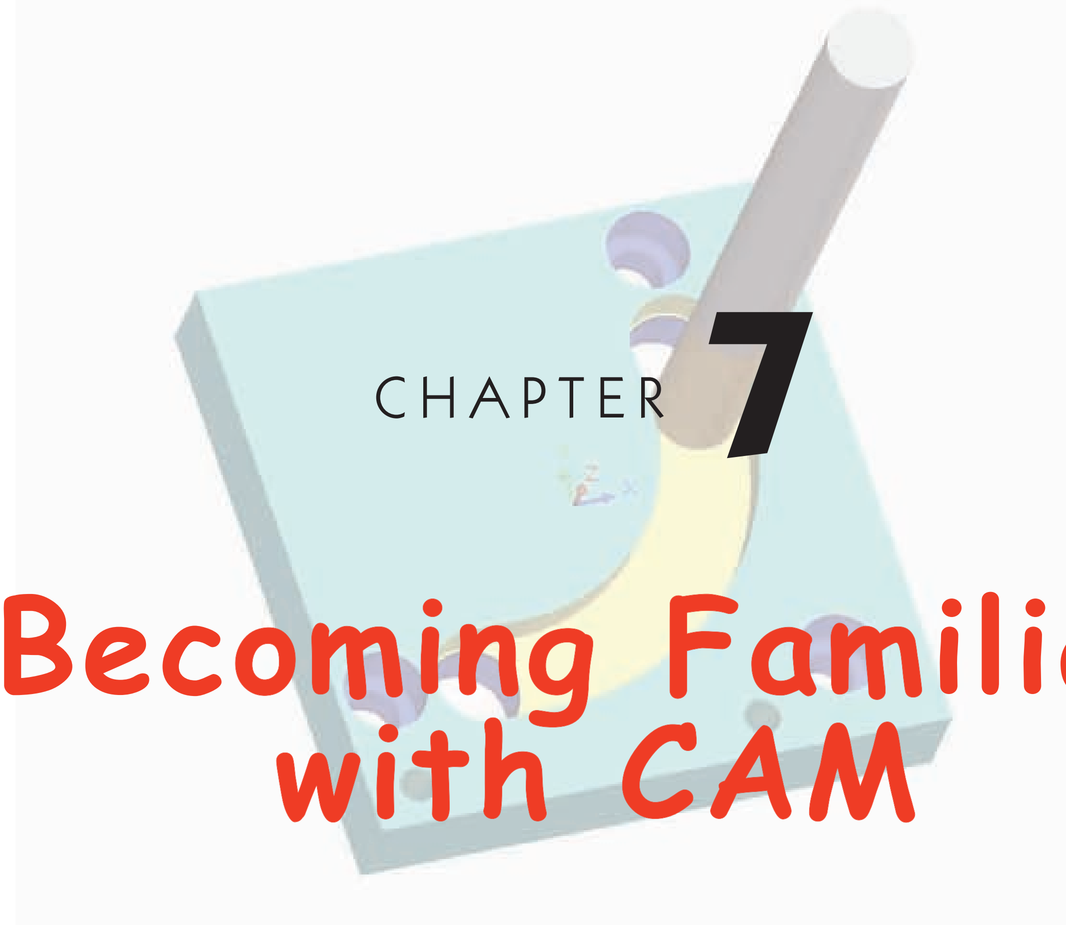


Figure 6-26 Indexing parts for machining can be challenging. The circular part on the left needed holes installed on the perimeter that are related to holes in the base of the part. An indexing feature, showing on the right, was designed, and an assembly made with the parts to show that the fixture would, in fact, index properly.

Suggestions for Becoming Familiar with CAD

1. Often the best way to start is simply to start.
2. Paying attention to detail and working methodically often leads to positive results.
3. Stick with an application that is comfortable for you.
4. People have different ways they like to learn.
5. The first step is to create a computer solid model.
6. Launch the main modeling screen.
7. Begin modeling by choosing a plane.
8. Develop the basic shape of the adapter plate.
9. Dimension and extrude the rectangle that represents the adapter plate.
10. Once the overall shape of the adapter plate is complete, focus on creating the holes.
11. Apply the corner radii to complete the model.
12. Practice constructing a few perplexing solid models.
13. Assemblies can be useful when designing indexing fixtures for machining parts.

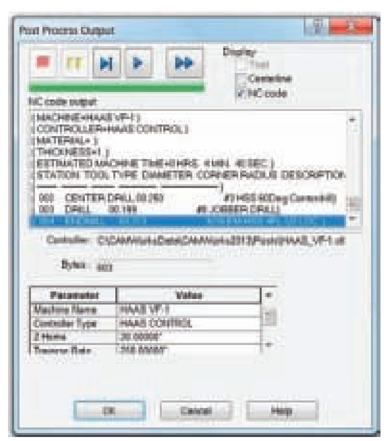


CHAPTER 7

Becoming Familiar with CAM

Our computer system recently crashed and was down for a few days. We were forced to write our programs manually to keep the wheels of industry turning. That was an eye opener! A job that would normally take minutes to program took over ten times as long, if you could even do it. And when a program was finally written manually, there was little confidence in it for good reason. It was usually loaded with mistakes.

The CAM system we use is Camworks®. We chose the software some years ago because it was the only CAM software at that time that could be integrated with SolidWorks. It has worked well for us. One of the advantages of having the software integrated with SolidWorks is that one you launch the SolidWorks application, the CAM application is immediately available for use. Another benefit of having the software integrated is that when working with a SolidWorks model, the tool paths applied to the model with Camworks are saved with the SolidWorks file. It is a compact, user-friendly ar-rangement.



Before I go on about using CAM (computer aided manufacturing) to make our lives easier, I like to point out that there are still jobs I feel more comfortable doing in conventional machines. Jobs where the setup is skewed in some way come to mind. We don't have a 5-axis machine; therefore, we have to tilt parts, which makes it difficult to pick up a starting point.

We ran a job a while ago on a CNC mill, putting vent pin holes in about a dozen mold cavities. The cavities had to be tilted to get the holes in at the correct angle. We mounted a tooling ball on the cavities at a known location to get a starting point. The holes in the cavities were relatively deep and difficult to reach. They had to be close in diameter and precise in depth for the vent pins to press into. The entire operation was risky in my opinion.

The next time the job came up, I chose to install the vent pin holes using a conventional milling machine (see Fig. 7-1). What a relief! The holes were installed in half the time it took to do them in the CNC machine, and with a lot less stress. The mold cavities were, of course, machined in a 3-axis CNC machine.

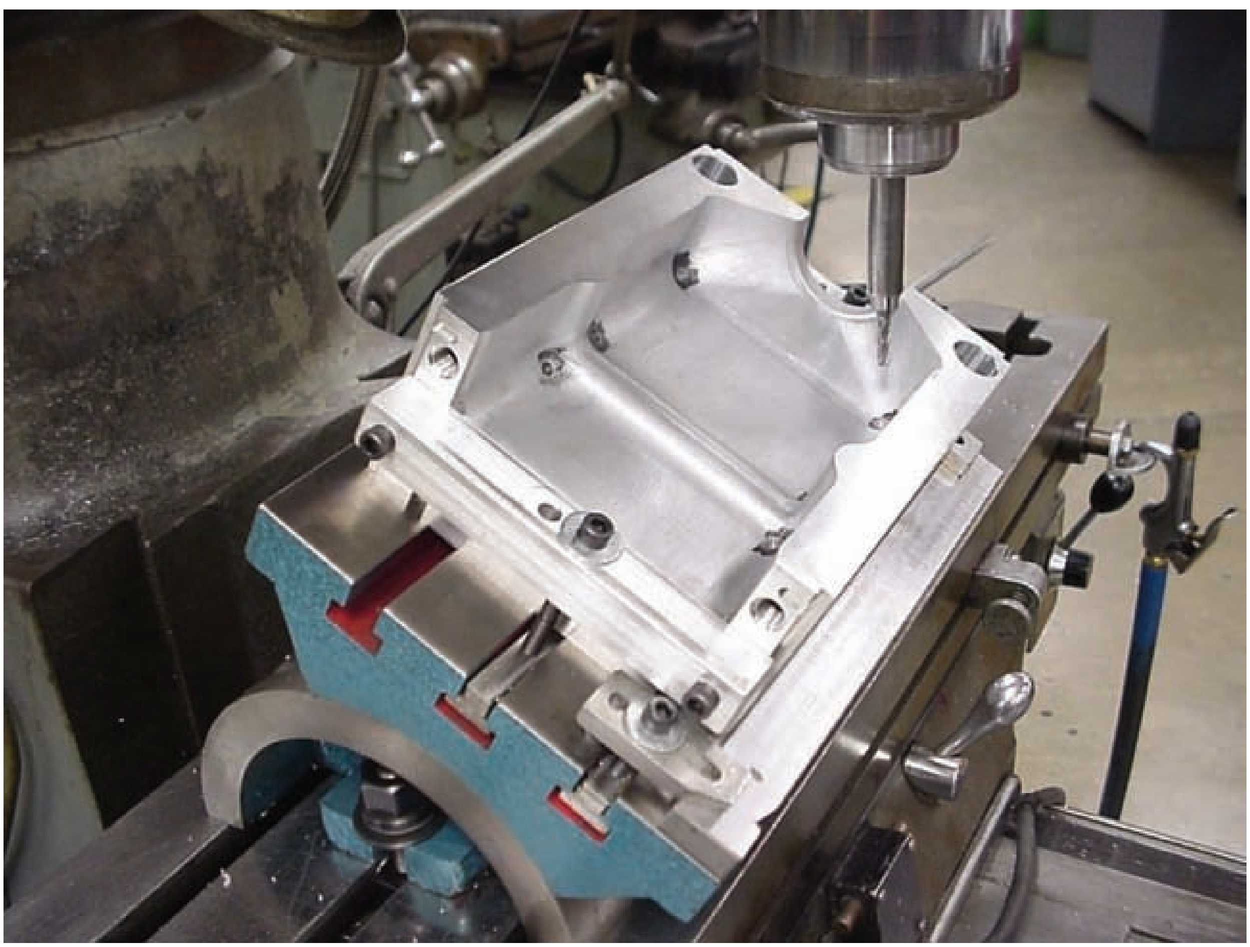


Figure 7-1 Cutting vent pin holes using a conventional milling machine.

Figure 7-2 shows another job I chose to run in a conventional milling machine. Again, these were holes that had to be machined in at angles. In this case, the angle was fairly extreme. The base part, however, was made in a CNC mill.

1. CNC machines are not suited for all tasks.

In many cases, CNC machines simply are not substitutes for competent manual machinists. Figures 7-1 and 7-2 show a couple of these examples.

2. The first thing you need for CAM is a computer model of a part.

The model allows you to apply cutter sizes and other machining parameters such as feeds, speeds, and depth of cut to simulate cutting the model on the computer. Using CAM, you can get a good but not perfect idea on a computer about how the process will run before putting the job on a machine.

3. CAM applications will generate the G-codes a machine uses.

The G-code tells the spindle where to move and how fast to cut using a particular cutter. However, generating G-code is almost an afterthought because it takes no thinking on your part. Once you've chosen tool paths and input machining parameters, just a few clicks will output the G-code.

For this discussion, I am going to use CamWorks ™ because it is what I use. Other CAM applications will have different screens, methods of inputting machining parameters, and ways of generating tool paths. Regardless of the interface, all CAM applications will have ways to choose and input how you want to machine parts. My apologies to the CamWorks people if I under-represent the capabilities of their software. This is just a cursory introduction to give readers an idea of how to use CAM. I've been successful making thousands of different parts using this software.

Because I am primarily a machinist who needs to get jobs done, I tend to use strategies that are simple and easily managed. Breaking programs down into manageable segments makes editing and troubleshooting easier.

Figure 7-2 Cutting holes into a tilted part using a conventional milling machine.

4. Automatic programming utilities can be handy, but you'll still fine tune the program to get things to run the way you want.

I tend to shy away from automatic programming utilities at the expense of taking more time initially. I prefer choosing the types and sizes of the cutters myself.

At "tool shows," where vendors demonstrate CAM software, they can program just about anything in seemingly no time at all and the part will cut beautifully on the computer. But you can bet vendors who demonstrate machines actually cutting metal have spent a lot of time fine tuning their programs.

5. Camworks is divided into two sections: a features side and an operations side.

These sections are shown on the main SolidWorks screen. The features side is used mainly for defining or outlining the geometry of the features you want to cut. The cutting (or operations) side is used for inputting cutter sizes and inputting cutting parameters such as feeds, speeds, and depth of cut.

6. Begin the programming process by clicking the left Camworks icon in the main SolidWorks screen. (see Fig. 7-3)

Once you click the icon, you are entering the CAM application, but not closing the SolidWorks application. You can then work on the features side of the software, where you will define the features you want to cut.

Figure 7-3 You'll likely go back and forth between Camworks and Solidworks as you program parts.

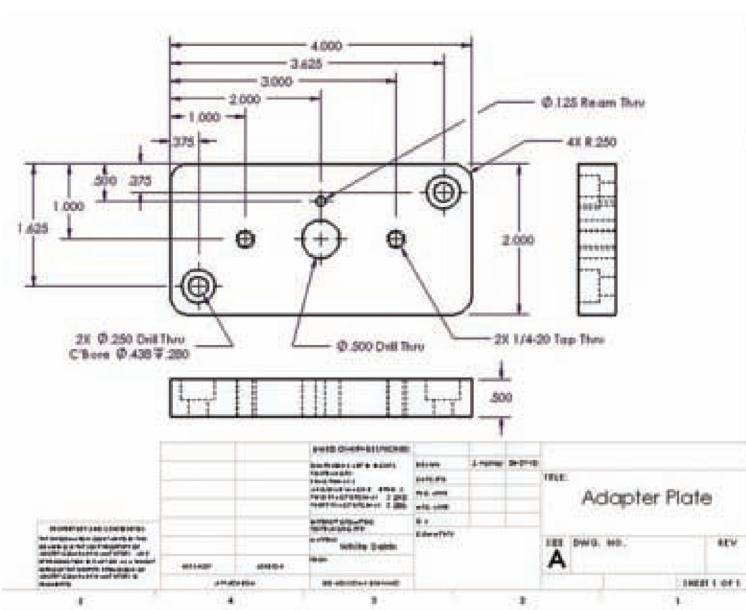


Figure 7-4 Multiple views of the adapter plate. This is the same drawing seen in Figure 6-4.

7. The toolpaths applied to machine this adapter plate are relatively simple and commonly used.(see Figure 7-4)

Any CAM application within reason will allow you to apply the same toolpaths. It becomes a matter of learning how to manipulate your CAM application.

8. Insert a part setup. (see Fig. 7-5)

A part setup in a CAM application represents how you would set up a part in a machine. With a 3-axis vertical milling machine, the part setup is the same as a setup in a manual vertical milling machine.

Working in the features side of Camworks, insert a mill part setup by clicking on the top face of the model. This allows the cutting operations, including the counter bores, to be machined in one setup.

An arrow appears on the face of the adapter plate showing which way cutters will approach the model and the actual part in the machine.

The adapter plate in Figure 7-5 will be machined in two setups. The first will cut the holes and the perimeter of the part. The second will machine the excess material off the back side. The planning for this adapter plate is uncomplicated because there are only two setups. That won't always be the case. The more setups needed to machine a part, the more thought you must put into planning.

Much of the thought you put into planning is to determine how you are going to hold the part for each setup, in a way that provides access to the features you want to cut. You may also have to plan for cutter runoff.

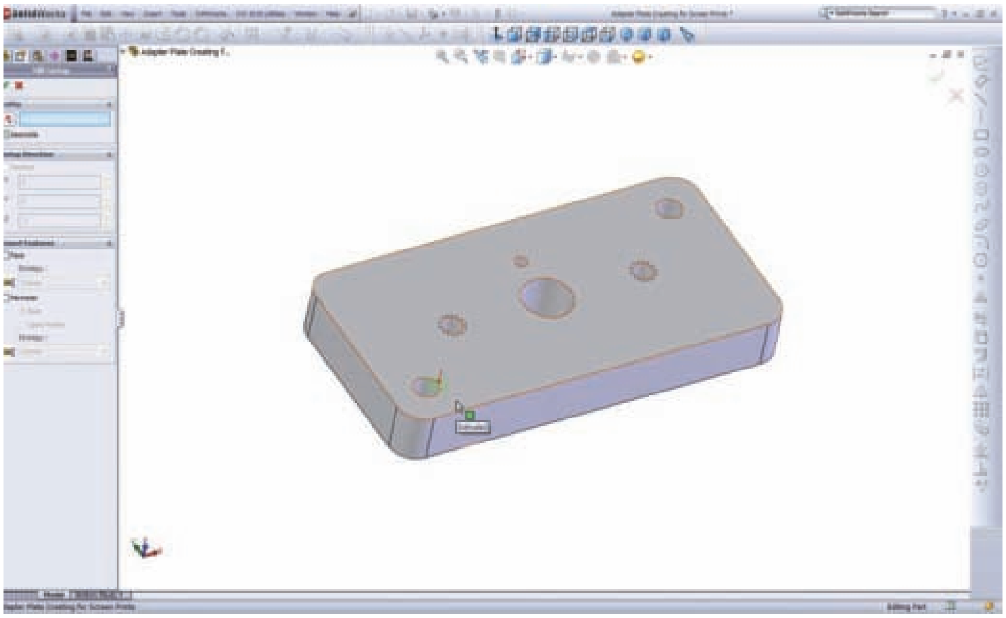


Figure 7-5
A part setup is defined showing which way cutters will approach the model and the part when it is machined.

9. Select the features you want to cut for Setup #1 (see Figs. 7-6, 7-7, and 7-8)

In this setup, we want to machine the holes and the perimeter of the part. The first thing the system needs to know is the geometry of the features you want to cut.

There is a utility in Camworks that automatically extracts the machinable features, but we're not going to use that here. Instead, we are going to pick each feature separately so you are controlling right from the beginning the features you want to cut.

Most of the features in this adapter plate are holes. The only non-hole feature is the part's perimeter, which Camworks defines as a boss.

There are only two types of features you will ever cut: 2D or 3D. Camworks defines these as 2.5 Axis Features and Multi-Axis Features. With this adapter plate, there are no 3D or Multi-Axis Features that need to be cut.

Let's begin by defining the "hole" features. Click Insert 2.5 Axis Feature in the menu. A screen opens that allows you to choose the type of feature you want to insert. These are holes so we choose "holes."

Click the bottom edge of each hole in the adapter plate including the bottom edges of the counter bored holes and the tapped holes.

A menu opens that allows you to choose the extent of the features. Click "Up to Face" in the menu; then click the top face of the adapter plate. Click finish. Once you do this, you're done defining the holes. The system now knows everything it needs about the holes, including their diameters and depths.

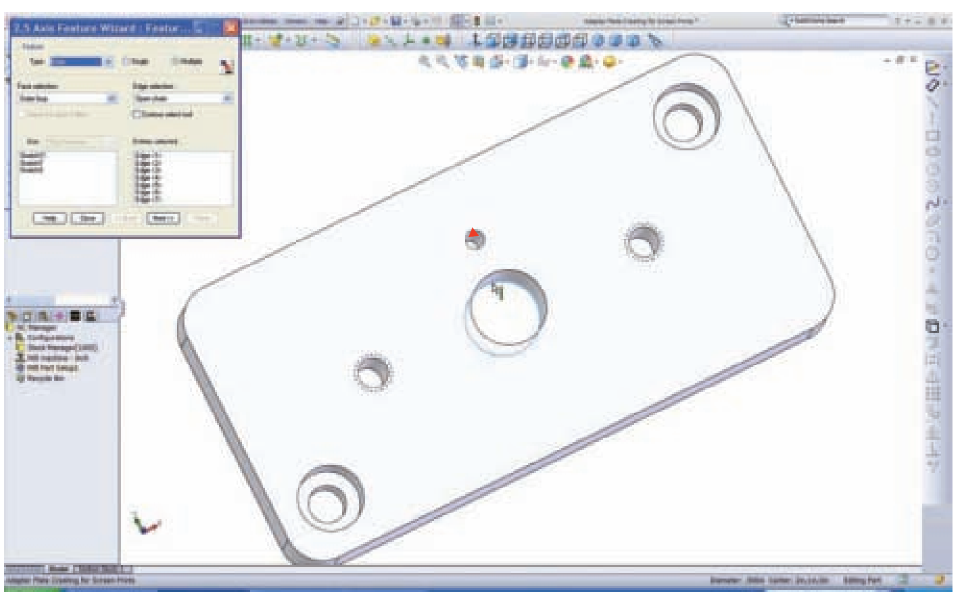


Figure 7-6 "Hole" features are being selected and defined here. Once selected, the features will reside in a list on the features side of Camworks. Cutting parameters such as feeds and speeds can then be applied later to any features in the list. Cutting parameters are input on the operations side of Camworks. The arrows in this figure point to the edges that should be clicked on.

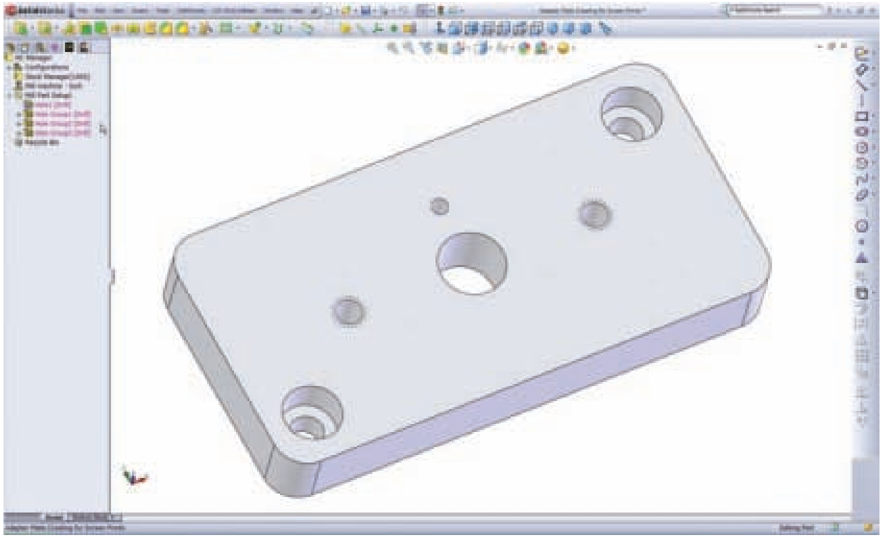


Figure 7-7 Once chosen, the features to be cut are listed in the features tree on the left side of the screen.

Camworks conveniently separates the holes into groups that can be seen in the features tree (see Fig. 7-7). Each group contains features that are exactly alike.

The only other feature we need to define for Setup #1 is the perimeter of the part. In Setup #1, insert another 2.5 Axis Feature which, this time, is the perimeter of the part. The perimeter of the part is defined as a Boss in Camworks. Click Boss in the menu; then click the bottom perimeter edge of the adapter plate (see Fig. 7-8). Now we need to define the extent of the boss.

In the menu that opens for defining the extent of a feature, choose "Up to Face." Then click the top face of the model. Finally, click Finish and you are done defining the boss. The system now knows everything it needs about the boss, including its dimenions and its thickness. The boss feature as well as the hole features appear in the features tree.

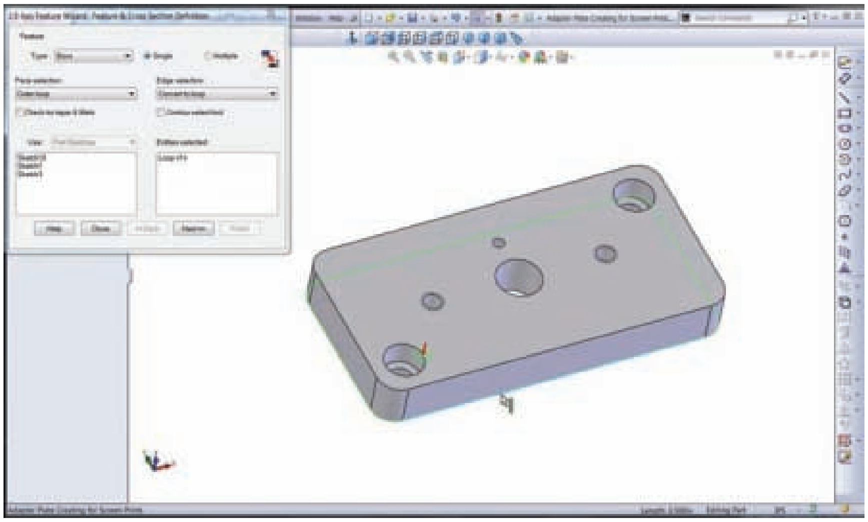


Figure 7-8 The remaining feature to define for the system is the Boss feature, or perimeter of the part. There are a few ways to do this, but the simplest is to click on the bottom perimeter of the part. Then in the dialogue box, click the top face of the part to define the feature's thickness or extent.

10. Once all the features have been defined for the first setup, jump over to the operations side. (see Fig. 7-9)

In the case of the adapter plate, once the holes and perimeter have been defined, use the Camworks operations icon to move to the operations side of the application.

A screen appears that shows Setup #1 on the operations side. It is empty because we haven't added any tools yet. You can now start choosing tools and inputting various cutting parameters such as speeds, feeds, and depth of cut for each tool you choose.

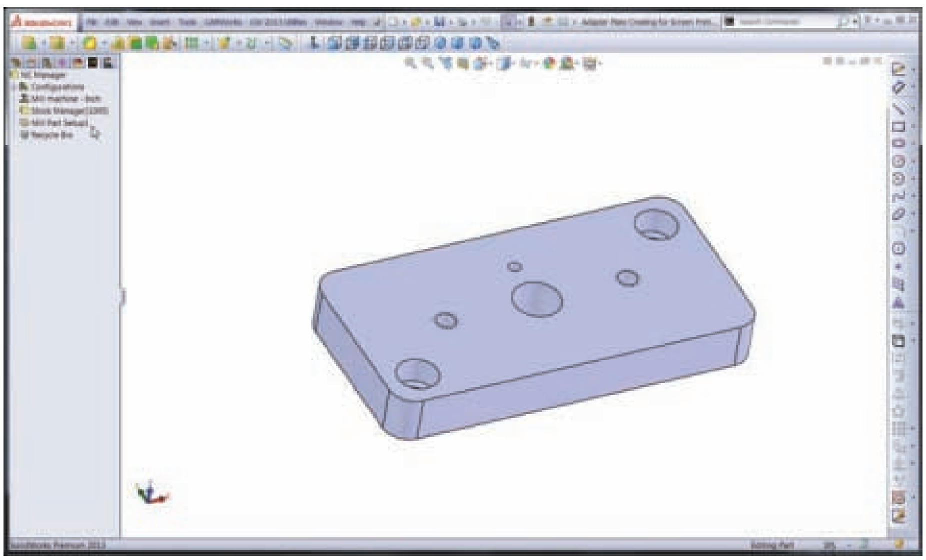


Figure 7-9 Once in the operations side, tools and cutting parameters can be input.

11. Select your tools and their parameters. (see Figs. 7-10 through 7-20)

Once you select a tool, the main screen for inputting these parameters pops up. For example, Figure 7-10 shows the screen for a center drill. There are tabs at the top of the screen you can select between to input cutting parameters.

For each tool you'll be using, the system will want to know the spindle speeds, the feed rate, the allowance amount for roughing when using end mills, the depth of cut, the lead in and lead out (when applicable), and the number of "spring cuts" for finishing.

In essence, what you are doing with CAM software is first choosing, and defining the features you want to cut, and then defining how you want to cut them.

Once you learn your software, you'll be able to input this information easily. Therefore, I won't be going into extensive details on the screens used to input this information. It's more important that you master the broader concepts; the specific screen details may vary as new versions of the software come out.

The first tasks we need to do are to center drill the holes. (Figure 7-11).

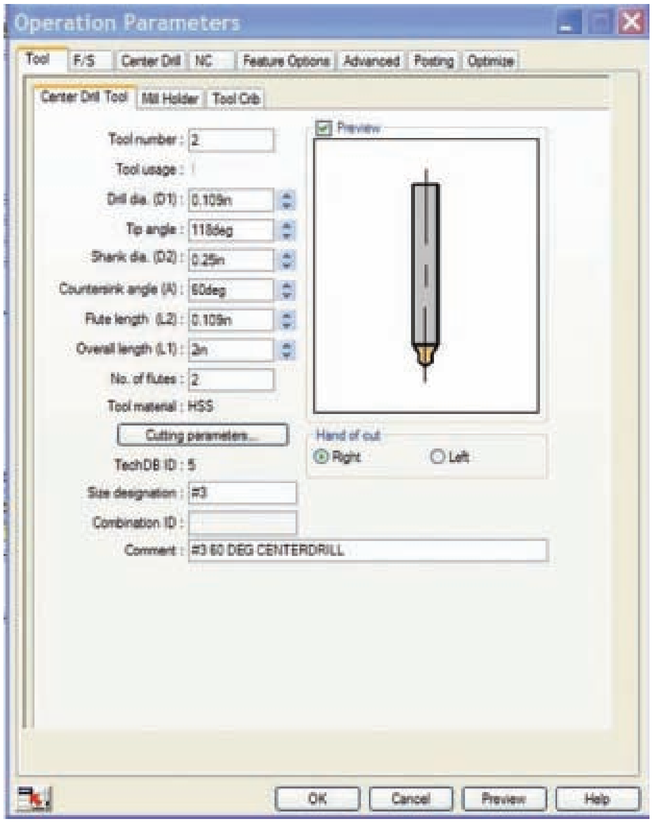


Figure 7-10 In this case, a center drill is selected for tool #1.

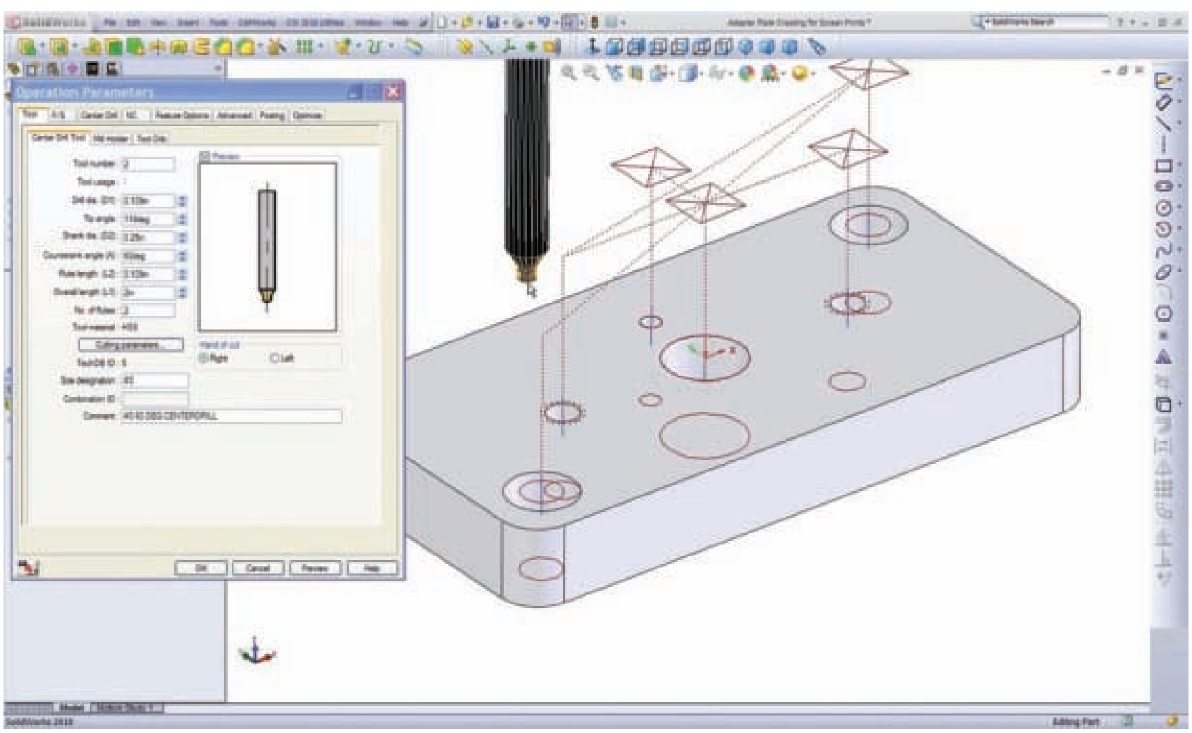


Figure 7-11 Center drill.

In Figure 7-12, a .120" diameter drill was chosen to drill the pilot hole for the 1/8" reamer, which follows in Figure 7-13. Drilling and reaming thru holes is easier than drilling and reaming blind holes. With thru holes, you don't have to concern yourself with drill and reamer depts., chip packing, or coolant flow through and around the reamer.

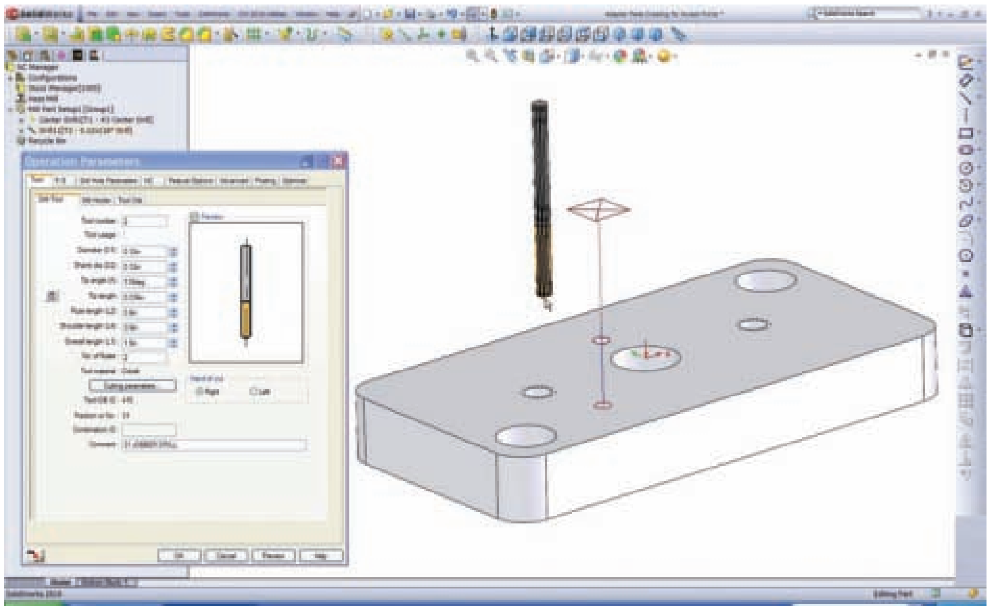


Figure 7-12 This undersize pilot drill is followed by an 1/8" reamer.

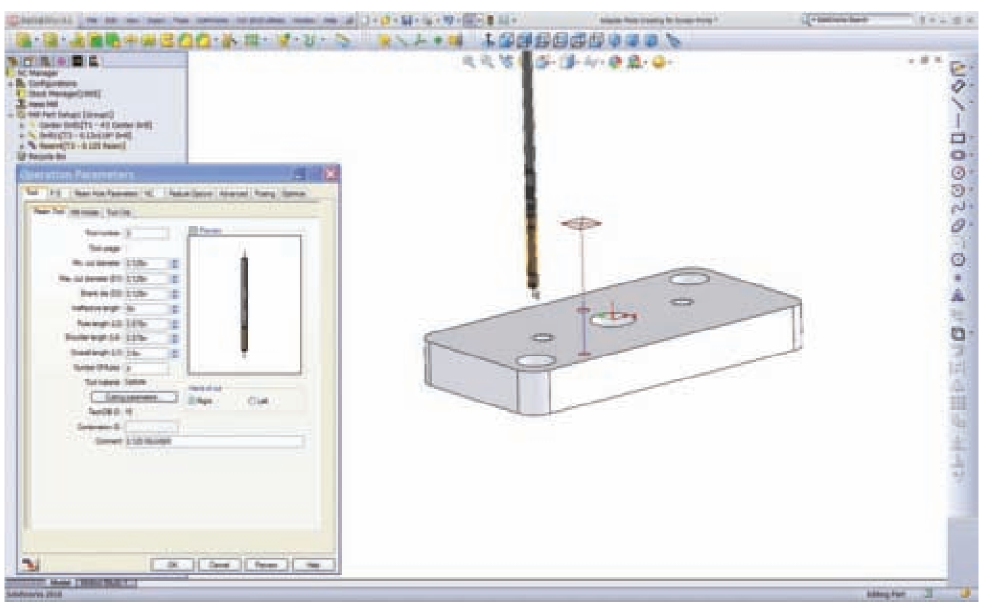


Figure 7-13 1/8" reamer.

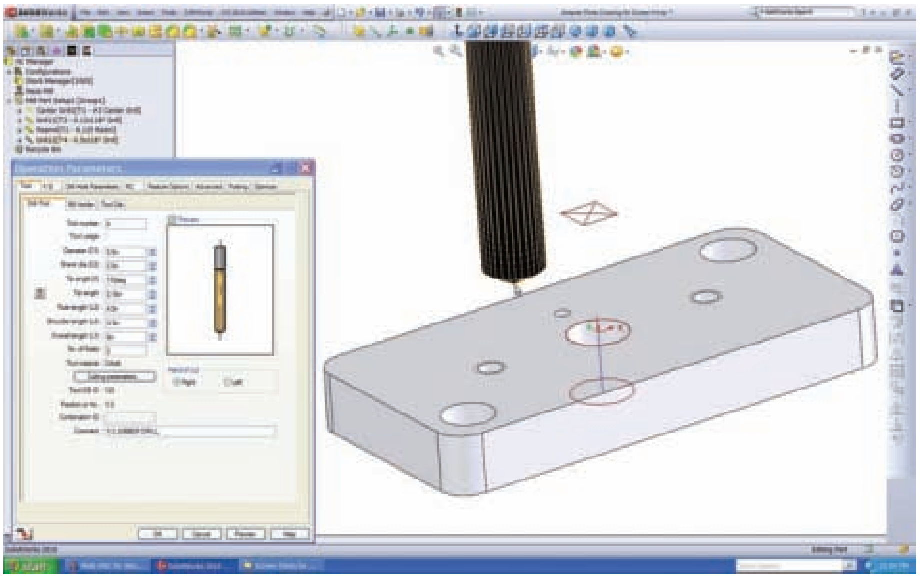


Figure 7-14 1/2" drill.

A 1/2" diameter drill was chosen from the Camworks tool crib to drill a hole in the center of the plate (Fig. 7-14). Next, a 1/4" diameter drill was chosen to drill the two clearance bolt holes in the corners of the plate (Fig. 7-15). Normally, a .265–.281" diameter clearance drill would be used for 1/4" bolts. In some instances, on-size clearance holes for bolts can be used to provide close part positioning without having to use dowel pins.

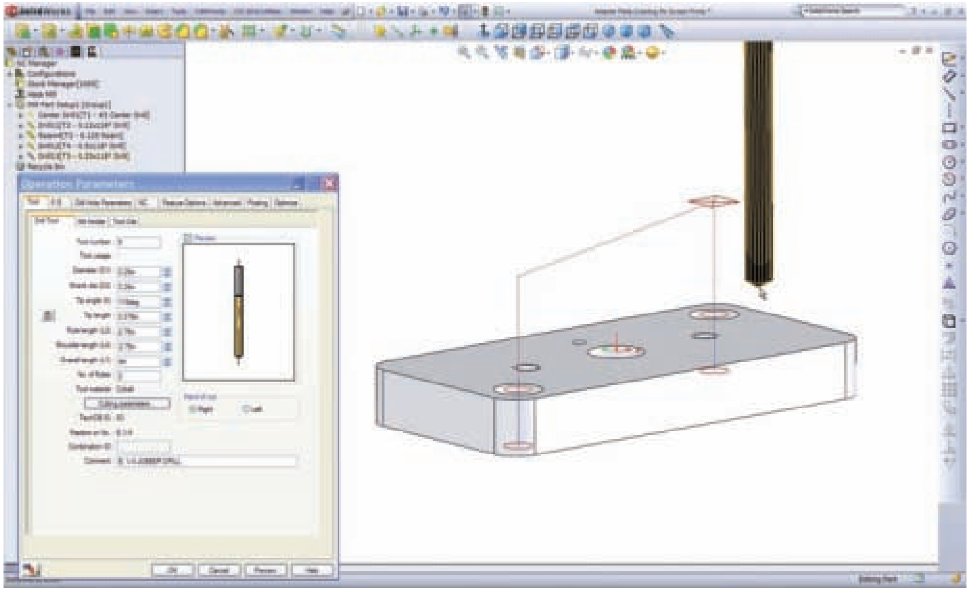


Figure 7-15 1/4" drill.

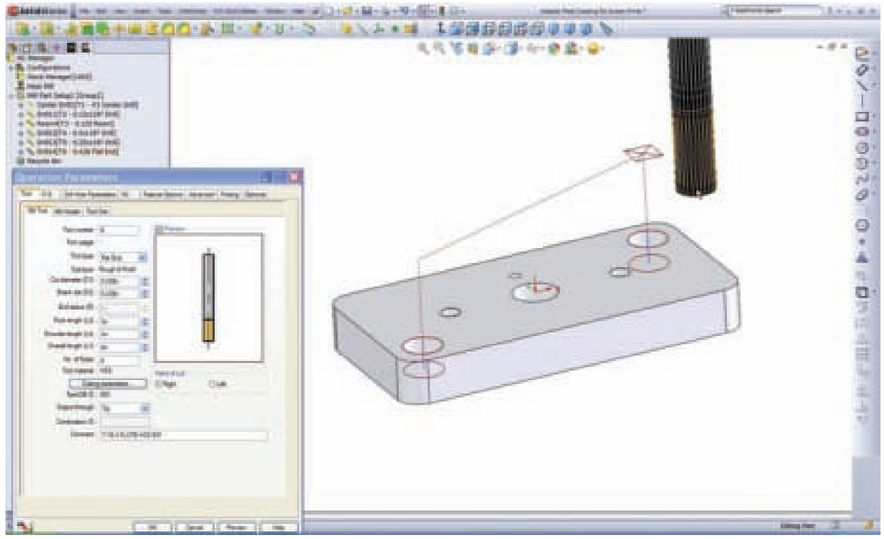


Figure 7-16 7/16" end mill for counter boring.

Other tasks that are part of preparing the adapter plate are counter boring the bolt clearance holes (Figure 7-16) and tap drilling the two 1/4 -20 holes (Figures 7-17 and 7-18). Notice how the operations tree grows as more tools are added.

As a side note, it's a good idea to save computer work often. If your computer locks up and you haven't saved your programming, you'll be obliged to start over. Sometimes, when our computer isn't feeling well, I'll save the file after programming each feature. Saving the file is especially important when you start programming long 3D cuts. There's a lot of computer processing going on when programming 3D cuts, and glitches are not uncommon.

The remaining operation to be completed for Setup #1 is to cut the part's perimeter. A 1/2" diameter end mill was chosen for this operation, as shown in the operations tree as the last tool in Setup #1 (Fig. 7-19). Other diameter end mills could have been chosen,

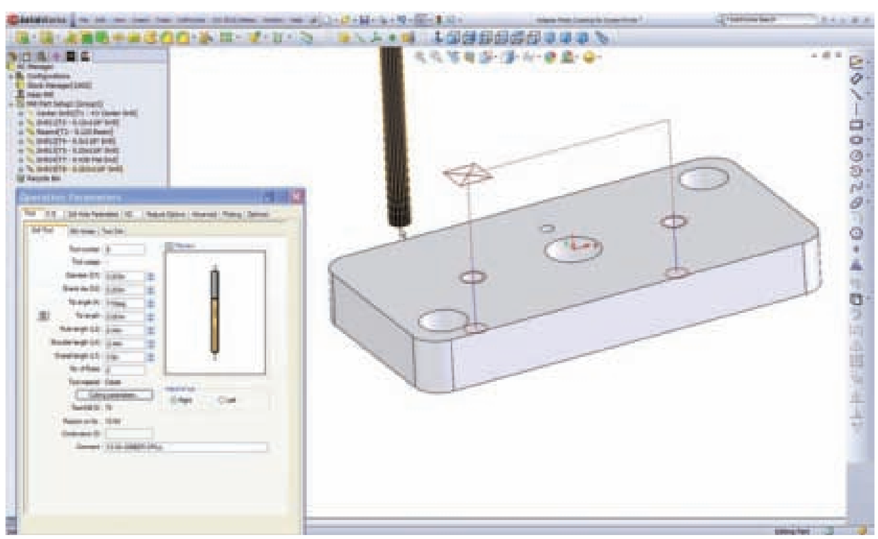


Figure 7-17 Tap drill for 1/4 - 20 holes.

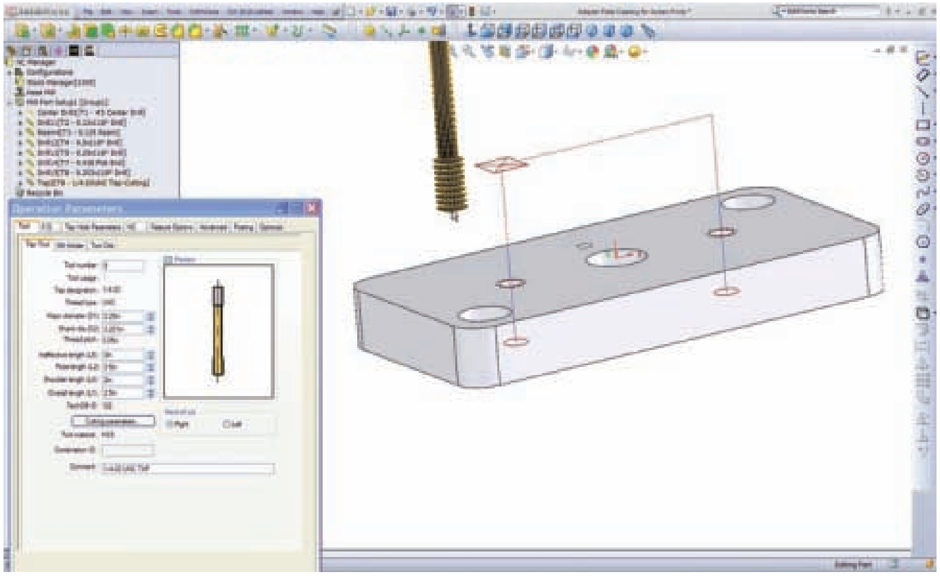


Figure 7-18 1/4 -20 tap.

but 1/2" diameter end mills are common, plenty strong for this cut, and there's likely a 1/2" end mill already mounted in a tool holder at the machine, which would make setting up this job easier.

As a side note, in this example, the 1/2" end mill was programmed to go right to size to finish the overall length and width. This is likely sufficient for a soft material like Delrin. However, if the material was something tougher, like stainless steel, I would rough in the perimeter first with a roughing end mill, leaving a few thousandths of material per side, then come back with a fresh (sharp) end mill, and take a final pass to bring in the overall length and width.

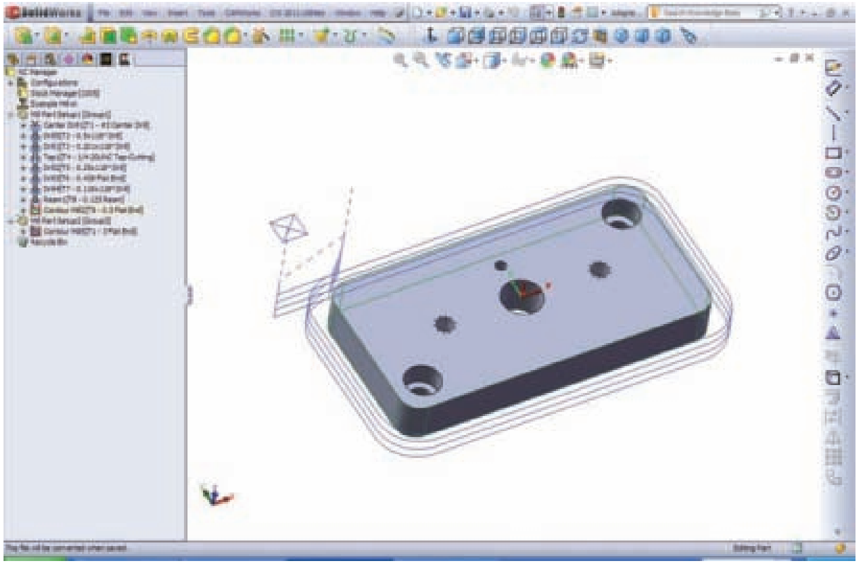


Figure 7-19 Perimeter tool path for 1/2" diameter end mill. Note the lead on and lead off for this cut.

12. A standard CAM feature helps you optimize your cutting time. (see Fig 7-20)

A really nice feature of CAM software is that you can see how much time a cut takes. Furthermore, you can see how much cutting time a setup or a complete part takes. When inputting cutting parameters, you'll often be surprised by how much or how little time is needed for an operation. Then you can make adjustments accordingly. Figure 7-20 shows how much time it takes to center drill the adapter plate at the input feed rate.

The only thing we do in Setup #2 is face the back side of the part to the correct thickness. A 3" diameter end mill was chosen to complete this facing operation.

13. Simulate cutting the part on the computer. (see Figure 7-21)

Once all the tools and cutting parameters have been entered, you can and should simulate cutting the part. Rendering is of great value. It not only shows that the proper tools have been selected, but also it shows any potential interference that causes crashes. Figure 7-21 shows an example of a mold cavity cut rendered on the computer.

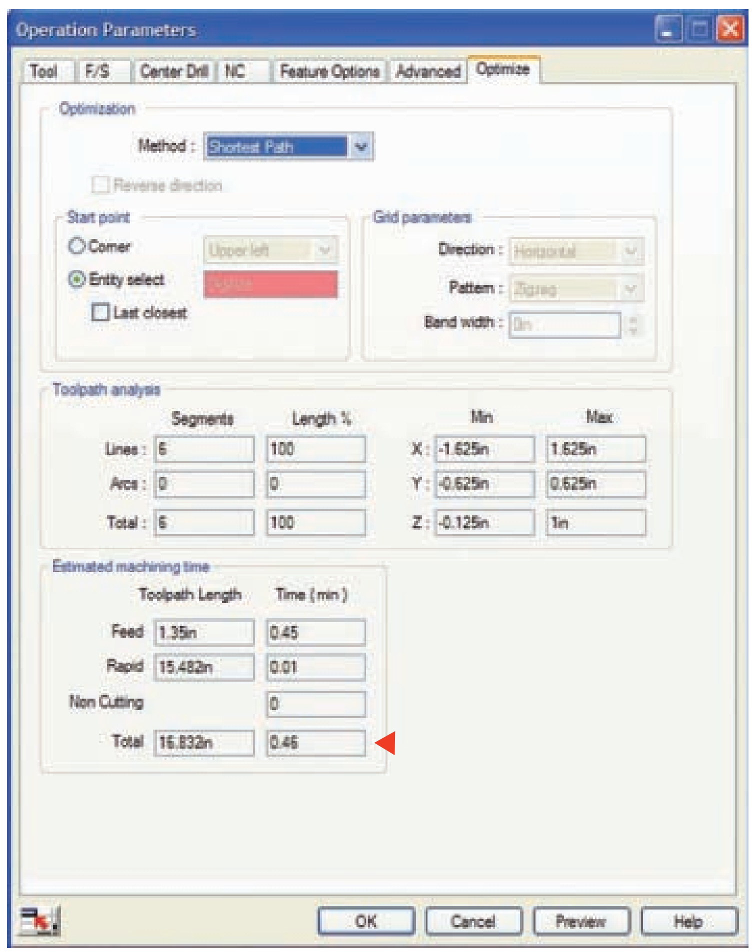


Figure 7-20 In this example, the cutting time for the center drill was .46 minutes. This is handy information. You'll often be surprised by how much time or how little time a cut takes. Then you can make cutting parameter adjustments accordingly.

14. Output the G-code.

Now that all the tool paths have been defined and shown to run smoothly, the only thing left to do is output the G-code, or CNC program, which the CNC machine uses to cut the part. This is call post processing.

Once the tool paths have been post processed into G-code for your particular machine, you can send the program over to your CNC milling machine via communications software or, less preferable, save the G-code to a disk which some older machines accept.

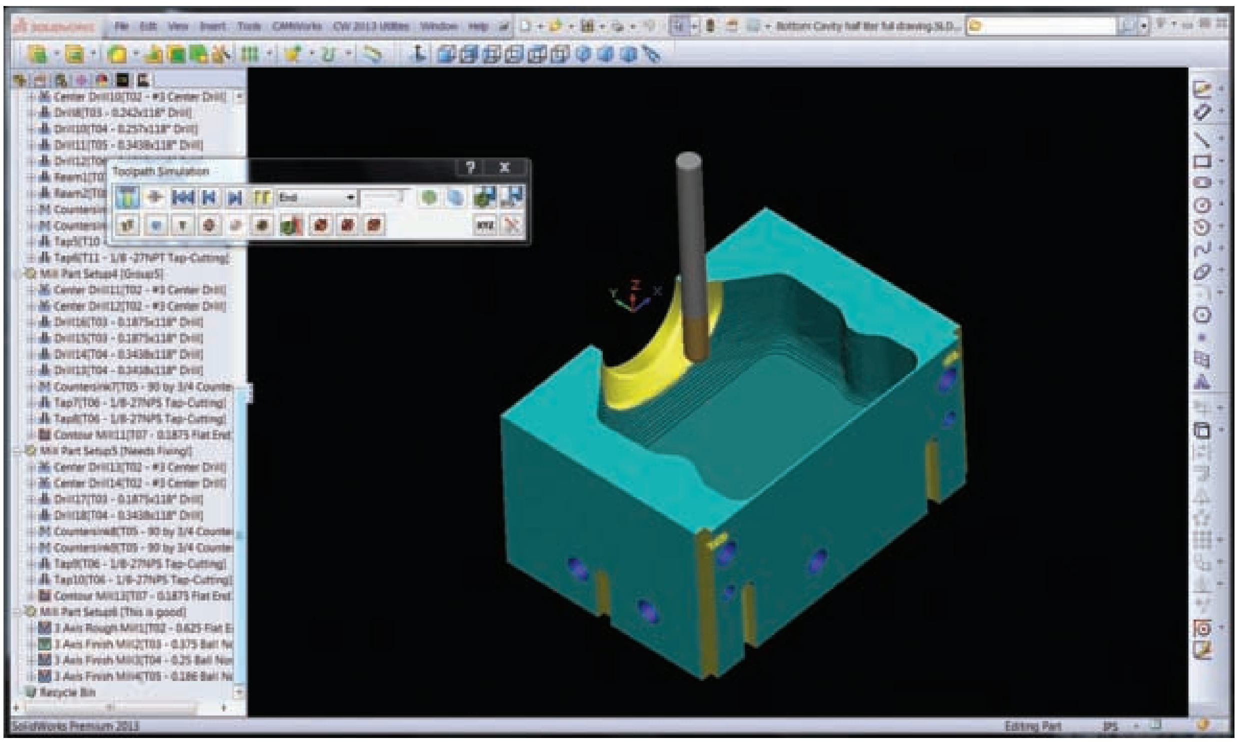


Figure 7-21 This computer rendering shows a ball nose end mill in the process of cutting a 3D mold cavity.

Suggestions for Becoming Familiar with CAM

1. CNC machines are not suited for all tasks.
2. The first thing you need for CAM is a computer model of a part.
3. CAM applications will generate the G-codes a machine uses.
4. Automatic programming utilities can be handy, but you'll still fine tune the program to get things to run the way you want.
5. Camworks® is divided into two sections: a features side and an operations side.
6. Begin the programming process by clicking the left Camworks icon in the main SolidWorks® screen.
7. The toolpaths applied to machine this adapter plate are relatively simple and commonly used.
8. Insert a part setup.
9. Select the features you want to cut for Setup #1.
10. Once all the features have been defined for the first setup, jump over to the op erations side.
11. Select your tools and their parameters.
12. A standard CAM feature helps you optimize your running time.
13. Simulate cutting the part on the computer.
14. Output the G-code.

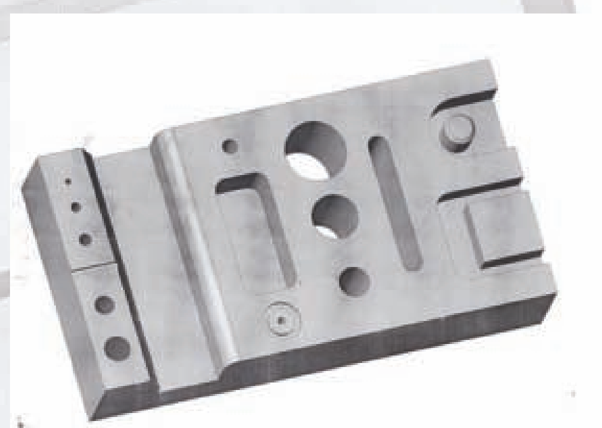
CHAPTER 8

Becoming Familiar with CODE

With the advent of CAM software, generating CNC or G-code is no longer the big deal it was a few decades ago. A CAM system will generate all the G-code you need in a matter of seconds.

"G-code" is an all-encompassing, informal term used when referring to CNC programming code. For example, you may hear someone say when trouble shooting, "the drawing is correct; the problem must be in the G-code." This means the problem is in the CNC program. Not all controllers use the same G-code, but generally there will be many similarities.

Learning G-code may seem tedious at times, but familiarity with it is necessary to edit programs at the machine. As good as CAM systems are at spitting out G-code, if you don't know the functions of the code and how to edit, you will be at a disadvantage in terms of making adjustments at the machine. Many unproven programs have "issues" that must be fixed or fine tuned.



1. The majority of machining operations can be undertaken with just a handful of G -and M-codes.

The G and M codes presented in this chapter are commonly used in day-to-day machining. G-codes can loosely be described as tool positioning codes and M-codes as switches for actuating mechanisms such as the tool changer and chip auger, or for opening and closing valves to allow coolant or air flow. M-codes have other functions as well. If you are new to CNC programming, it may look complicated at first. However, you will soon see simple patterns that are used over and over again.

A standard version of G-code known as RS-274-D was approved for the United States in 1980. Other countries use other standards. Differences to the standard RS-274-D have been added by machine tool builders over the years. At one time, these variations caused compatibility issues. Often, code for one machine couldn't be used for machining centers made by different manufacturers.

Now that machining operations are created with CAD/CAM applications, G-code can be output for a specific machine using the appropriate post processor, which is just a computer file. Using the correct post processor, compatibility issues are not much of a problem anymore, at least for computers. Operators may still have issues with variations in G-code as they move from one machine to another.

2. Writing programs manually is a good way to become familiar with G-code. (see Fig. 8-1)

I have always preferred learning something by doing a project, so that's how I'm going to present this material. Let's write a program manually to make the adapter plate discussed in Chapter 7. This adapter plate is similar to one I made recently to adapt a new hydraulic valve to an existing molding machine.

Let's assume this part is made from some easily machined material so that we can concentrate on the function of the code itself.

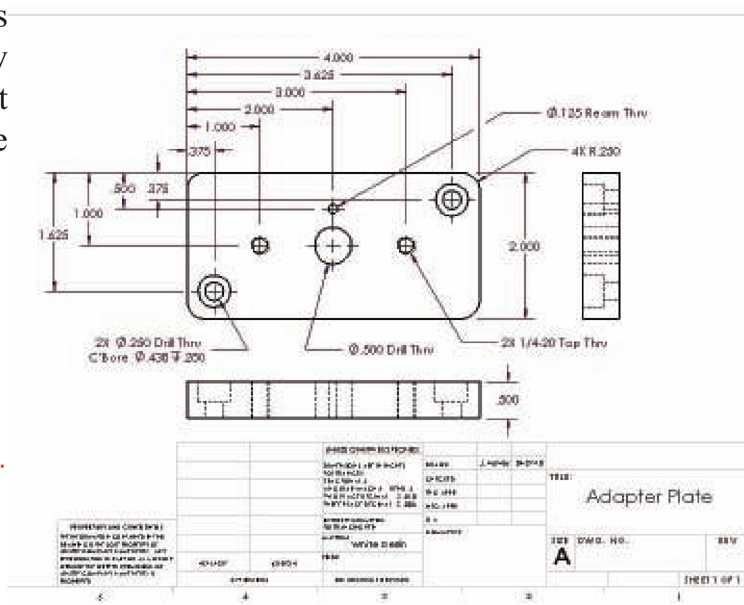


Figure 8-1 Adapter plate drawing.

3. Begin and end all programs with a percent sign.

%

The percent sign tells the controller where a program starts and ends.

4. Insert a program number and follow it with a description of the part and setup number in parentheses.

%
O00101 (Adapter Plate Setup #1)

Program numbers start with the letter O followed by a five-digit number. The program numbers can be whatever you choose. Note that the controller ignores anything in parentheses. Information inside of parentheses is for your benefit only.

Program numbers help the controller keep track of stored programs. Each program stored in the controller has a different program number. You can search for a program number, in the controller, entered the number into memory, and run it at will. Table 8-1 lists sample program numbers.

Table 8-1 Sample Programs

O00002	(Bar Pocket Cleanup)
O00003	(3F Slide engraving.nc)
O00005	(Manifold six hole milling)
O00009	((ADD SET Slide engraving.nc)
O00100	(PROBE TEST)
O00101	(ADAPTER PLATE DEMO SETUP #1)
O00103	(Filler Port Holder, 10 S)
O00129	(80038129 Setup)
O00191	(0191 Adjuster Support Setup #1)
O00201	(Hot Half Setup #1)
O00202	(FIX GATE)
O00204	(Hot Half Drill Hole in Plug)
O00205	(Hot Half Weld cleanup)

5. Insert safety line coding.

```
%

O00101 (Adapter Plate Setup #1)
G17 G20 G40 G49 G54 G64 G80 G90 G98
```

Safety codes reset registers in the controller back to a common condition so the program runs as intended. Some codes erase commands in the controller's memory that could potentially cause a program to stop running. Other codes place in the memory commands or information needed to execute the program as intended. Your controller is likely set up with a number of defaults, which would make some — if not all — of these codes redundant and unnecessary.

However, having a strong safety line at the beginning of your programs gives you peace of mind, knowing programs are going to start consistently. Furthermore, there is no execution time associated with using them.

G17 (XY Plane)
G20 (Inch)
G40 (Cancel Cutter Compensation)
G49 (Cancel Tool Length Compensation)
G54 (Start Point)
G80 (Cancel Canned Cycles)
G90 (Absolute Mode)
G98 (Initial Plane for Canned Cycles)

G90, for example, puts the machine in "absolute" mode, meaning that all cutter positions in the program are measured from the **G54** starting point. In this example, **G54** is set to the center of the part.

Your post processor, which is the software that spits out the code for your particular machine, can be set up to output whatever safety code you prefer. The ones shown on the previous page are what we use.

6. Call up a tool number, insert a tool change command, and follow in parentheses with a description of the tool.

```
%

O00101 (Adapter Plate Setup #1)
G17 G20 G40 G49 G54 G64 G80 G90 G98
T1 M06 (#3 Center Drill)
```

T1 is the center drill that will be installed and **M6** is the command that tells the machine to go ahead and install the tool. M06 functions the same as **M6**. In other words, the zero between **M** and **6** is meaningless. Similarly, **T01** functions the same as **T1**.

7. **Insert a spindle speed and turn the spindle on.**

```
%
O00101 (Adapter Plate Setup #1)
G17 G20 G40 G49 G54 G64 G80 G90 G98
T1 M06 (#3 Center Drill)
S2000 M3
```

S2000 sets the spindle RPM to **2000; M3** is the command that turns the spindle on clockwise. You can turn the spindle on counter-clockwise by using **M4**.

8. **Insert a "rapid" command and a starting point command; then tell the spindle where to go in X and Y.** (see Figs. 8-2, 8-3, and 8-4)

```
%
O00101 (Adapter Plate Setup #1)
G17 G20 G40 G49 G54 G64 G80 G90 G98
T1 M06 (#3 Center Drill)
S2000 M3
G0 G54 X0 Y0
```

G0 tells the machine to move fast. On our machines, the default rapid speed is 400 inches per minute when used at the 100% setting. We can override the rapid speed using the percentage buttons on the control panel, shown here in Fig. 8-2. When testing a new program, the rapid speed can and should be overridden to a slower rate.

Figure 8-2 The override buttons for the rapid speed are shown in the lower left corner of this controller.

G54 is the zero starting point from which all X, Y, and Z moves refer (Figure 8-3). In this case, we set the X and Y starting point to the approximate center of the part by locating the center with a pointer.

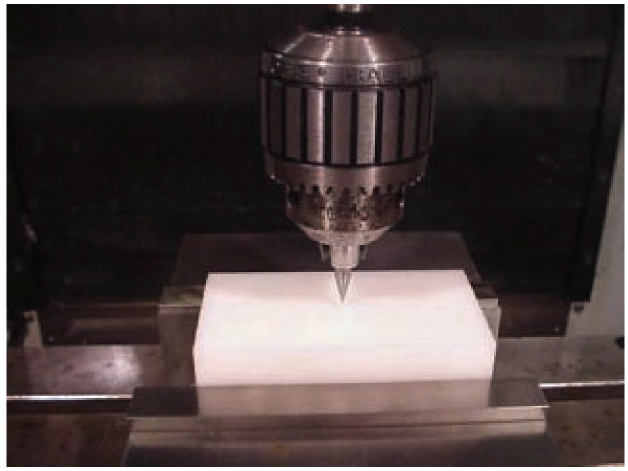


Figure 8-3 The X and Y starting point, or G54, is set roughly to the center of the part for this program.

There is extra stock on the sides to be cut away so the starting point in X and Y does not have to be precisely accurate. The pointer position is then input into the Work Offset screen in the **G54** register for X and Y (Figure 8-4). If more accuracy is needed for setting the starting point, then an edge finder, an indicator, or an electronic probe could be used.

We could have chosen a different starting point, but having the starting point at the center of the stock makes following the moves in this program easier.

If your starting point gets screwed up for any reason, you'll likely end up scrapping the part if you don't catch the error before cutting material. For example, if a machinist sets up the corner of the part as the starting point, when in fact the program was output to use the center of the part, then all cuts would be grossly mislocated.

X0 Y0 is the position the spindle is going to move to from the **G54** starting point. In this case, the first hole location is the same location as the **G54** starting point. When the controller reads this position, the spindle will move to **X0 Y0**, which is the center of the block, if it is not already there.

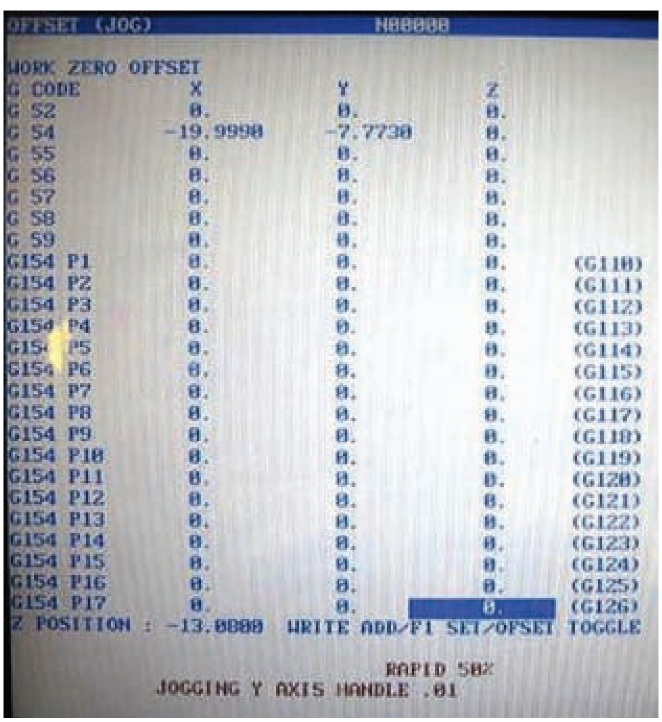


Figure 8-4 The X and Y starting point is entered into the Work Offset Screen once the position has been physically located on the part. In this example, we used a pointer to pick up the approximate center of the part. Because we are using G54, the X and Y locations are entered next to G54 on the screen. The values shown on the screen are of little significance to the operator. Those values are distances from Machine Zero and are values the controller uses to execute the program as intended.

9. Call up the tool length offset for T1, move the spindle in Z to the rapid plane, and turn on the coolant. (see Figs. 8-5 and 8-6)

```
%

O00101
(Adapter Plate Setup #1)
G17 G20 G40 G49 G54
G64 G80 G90 G98
T1 M06 (#3 Center Drill)
S2000 M3
G0 G54 X0 Y0
G43 H1 Z1. M8
```

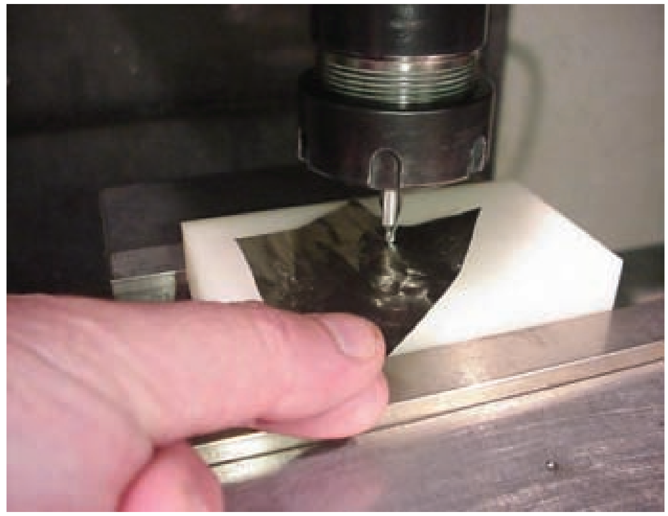


Figure 8-5 The center drill is being touched off to the top of the part. This Z position can then be entered into the Tool Length Offset screen.

G43 is a command that tells the controller to pick up a tool length offset.

H1 tells the controller from which register to pick it up. The tool length offset value is found by physically touching the tip of the center drill to the top of the part using a .001" shim (Figure 8-5).

All tools will have to be touched off this way and then entered into the Tool Length Offset screen (Figure 8-6). There are probes and gizmos that can do this automatically, but the shim method is what I prefer. Other people prefer using a probe. So be it!

Z1. is the rapid plane. Once **Z1**. is read by the controller, the tip of the center drill will move to one inch above the part in rapid mode. **G0** or rapid mode stays in effect until the controller reads a different feed rate. The rapid plane can be set to any height, but one inch above the part is a common default.

M8 is the command that turns the coolant on. Note that the coolant is turned on after the spindle has moved to the rapid plane, which is one inch above the part. There's no sense in turning on the coolant before that.

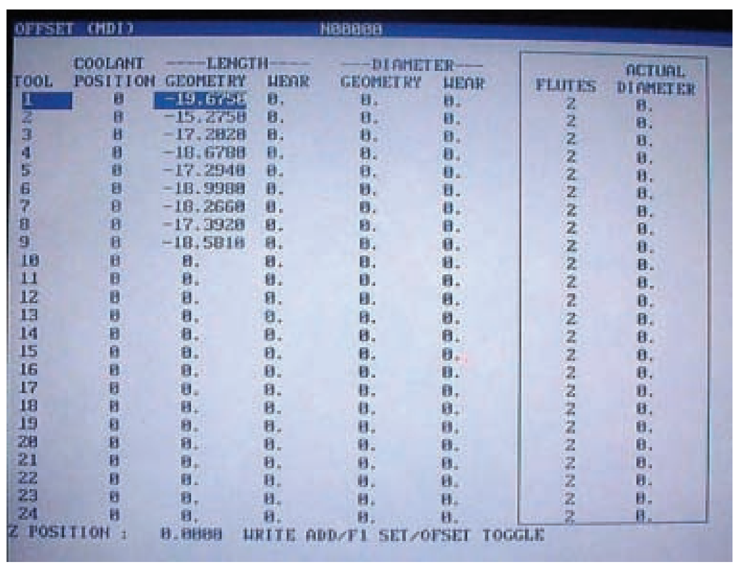


Figure 8-6 In this Tool Length / Diameter Offset screen, the values shown are tool length offsets. The tip of the center drill (T1) is –19.675" from machine Z zero. The nine values shown are for all the tools that will be used to cut the adapter plate.

10. Insert a drilling canned cycle to center drill hole locations.

```
%

O00101 (Adapter Plate Setup #1)
G17 G20 G40 G49 G54 G64 G80
G90 G98
T1 M06 (#3 Center Drill)
S2000 M3
G0 G54 X0 Y0
G43 H1 Z1. M8
G81 G98 R.1 Z–.125 F3.
```

A canned cycle is a line of code that sets cutting conditions for all equivalent features that follow that cycle until it is canceled.

Table 8-2 highlights the different elements from this line of code.

Canned cycles are user friendly in that you can adjust a lot of machining parameters for all equivalent features with just one line of code. (Lines of code are often referred to as “blocks” of code.)

Although commands in a canned cycle block can be placed in any order, the order shown in this example is typical.

The nice thing about canned cycles is that they can be easily edited. For example, if you want to change the feed rate or the depth of cut for all the center-drilled holes, you would simply alter whichever values you want to change within the canned cycle block of code.

Table 8-2 Center Drill Canned Cycle

Code	Explanation
G81	Begins a drilling canned cycle; commands the spindle to feed continuously at a commanded feed rate (in this case F3, or 3 in/min)
G98	Commands the spindle to retract to the Z1. rapid plane before doing a lateral move to a different hole location
R.1	The clearance plane from which the tool will start to feed down at the commanded feed rate
Z–.125	Commands the spindle to drill into the part to a depth of .125"; because the spindle is already directly over X0 Y0, that location will be center drilled once the controller reads the canned cycle block of code
F3.	The cutting feed rate for the center drill

11. Insert additional hole locations in X and Y for center drilling.

```
%
O00101 (Adapter Plate Setup #1)
G17 G20 G40 G49 G54 G64 G80 G90 G98
T1 M06 (#3 Center Drill)
S2000 M3
G0 G54 X0 Y0
G43 H1 Z1. M8
G81 G98 R.1 Z–.125 F3.
Y.5
X–1. Y0
X–1.625 Y–.625
```

```
X1 Y0
X1.625 Y.625
```

Once the controller reads a canned cycle block of code, the machine will use the same cutting parameters for each additional hole as called out in the canned cycle. In other words, the type of cut, (drilling) the clearance plane, the depth of cut, and feed rate will stay the same for each additional hole.

Y.5 is the second hole location to be center drilled. It is a half-inch directly above **X0 Y0.** Once the spindle is correctly located on an axis, that axis doesn't need to be called out in a program again for the spindle to move to the correct location. Hence, we can use **Y.5** by itself in the program to get to the second hole. However, you could use **Y.5 X0** for the second hole location in the program and things would work fine. A total of six hole locations will be center drilled with the code shown above.

12. Cancel the canned cycle, return to the "rapid" plane, and turn off the coolant.

```
G80 Z1. M9
```

G80 cancels the drilling canned cycle. **G80** is used to cancel all canned cycles regardless of what type they are. Once **G80** is read in a program, the controller is ready to move on to do other business.

Z1. returns the center drill to one inch above the part in rapid mode,

M9 turns off the coolant.

Explanations and programming for T1 are now complete. Table 8-3 shows the complete code for this portion of the overall program (pertaining to T1).

Table 8-3 Code for T1

```
%
O00101 (Adapter Plate Setup #1)
G17 G20 G40 G49 G54 G64 G80 G90 G98
T1 M6 (#3 Center Drill)
S2000 M3
G0 G54 X0 Y0
G43 H1 Z1. M8
G81 G98 R.1 Z–125 F3.
Y.5
X–1. Y0
X–1.625 Y–.625
X1 Y0
X1.625 Y.625
G80 Z1. M9
```

13. Once the programming for T1 is complete, proceed with the programming for T2 and any additional tools. Describe each tool in parentheses.

Table 8-4 summarizes the code for T2. Note that this code is a portion of the overall program. Therefore, it neither starts nor ends with the percent % sign.

Table 8-4 Code for T2

```
T2 M6 (1/2 Jobber Drill)
S1400 M03
G0 G54 X0 Y0
G43 H02 Z1. M08
G83 G98 R.1 Z-.8 Q.1 F2.5.
G80 Z1. M09
```

T2 is the tool that will be installed — in this case, a 1/2 jobber drill — and M6 tells the machine to go ahead and install it.

Next, we insert a spindle speed **(S1400)** and turn on the spindle **(M03)**.

In the line that follows, we insert a "rapid" command (G0), a starting point command **(G54)**, and tell the spindle where to go in **X** and **Y (X0 Y0)**.

We then call up the tool length offset for **T2 (G43 H02)**, move the spindle in **Z** to the rapid plane **(Z1.)**, and turn on the coolant **(M08)**.

The next line is a peck drilling canned cycle that tells the machine to drill one 1/2" diameter hole. **G83** begins the peck drilling canned cycle. Peck drilling means the drill will repeatedly cut the amount called out by the **Q** value in the canned cycle that allows chips to escape.

G98 commands the spindle to retract to the "rapid" plane **(Z1.)** before doing a lateral move to a different hole location.

R.1 is also the clearance plane from which the tool will start to feed down at the commanded feed rate, which in this case is **F2.5. R.1** is also the plane to which the drill retracts after each peck, to give chips a chance to escape.

Z-.8 is the final depth to which the drill will cut.

Q.1 is the peck amount. In other words, the drill will feed .1" at the commanded feed rate for each peck into uncut material before it retracts to the clearance plane, which in this case is .1" above the surface of the part. The drill will retract and approach uncut material at "rapid" speed for each peck.

Cancel the peck drilling **(G80)**, return to the "rapid" plane **(Z1.)**, and turn off the coolant **(M09)**. Note that there is only one 1/2" hole in this part, which is located at **X0 Y0**. Hence, there are no additional hole locations called out after the peck drilling canned cycle.

This completes T2 and we can now proceed to T3.

14. Call up T3 and insert a peck drilling canned cycle to drill two .201 diameter ¼-20 pilot holes.

Table 8-5 summarizes the code for T3. As with T2, it is part of a larger program; therefore, it does not start with the percent sign %.

Table 8-5 Code for T3

```
T3 M06 (#7 Jobber Drill)
S1500 M3
G0 G54 X-.1 Y0
G43 H03 Z1. M08
G83 G98 R.1 Z-.9 Q.1 F3.
X1.
G80 Z1. M09
```

The **G83** peck drilling canned cycle used for this drill (T3) is essentially the same as the canned cycle for T2, which drilled the 1/2" diameter hole. The differences are the final Z depth and the feed rate.

The hole locations for this tool relative to the starting point are at **X-.1 YO** and **X1. Y0**.

15. Call up T4 and insert a tapping canned cycle to tap two ¼-20 pilot holes.

Table 8-6 summarizes the code for T4.

Table 8-6 Code for T4

```
T4 M06 (1/4-20 Tap)
S500 M3
G0 G54 X-1 Y0
G43 H4 Z1. M8
G84 G98 R.3 Z-.8 F25.
X1
G80 Z1. M09
```

G84 begins a tapping canned cycle also known as rigid tapping.

The **G84** rigid tapping canned cycle plunges a tap in and out of a tap drill size pilot hole. When using **G84**, the spindle feed rate and spindle RPM must be synchronized to match the pitch of the tap. Otherwise, the tap would break. CAM systems figure this out for you. When we write a program manually, we need to use a formula to get the synchronization correct.

Formula to Synchronize Spindle Feed Rate and Spindle RPM

Spindle RPM / Pitch of Tap = Feed Rate of tap in in/min

Using numbers from the T1 program, 500/20 = 25 in/min

If spindle RPM = 750, then the feed rate would be 750 / 20 = 37.5 in/min

R.3 is the clearance plane that is normally used for tapping holes. The tap will start feeding at the commanded feed rate and RPM from .3" above the hole. The .3 clearance plane gives the machine a little extra time to overcome inertia and synchronize properly.

Note that once this controller reads **G84** in a program, any manual overrides of either the spindle RPM or the feed rate by the control panel buttons will be ignored. The machine will always tap at the RPM and feed rate commanded in the tapping canned cycle block of code. This is a very nice safety feature to avoid broken taps.

16. Call up T5 and insert a peck drilling canned cycle to drill two 1/4" diameter holes.

Table 8-7 summarizes the code for T5.

Table 8-7 Code for T5


```
T5 M6 (.250 Jobber Drill)
S1400 M3
G0 G54 X1.625 Y.625
G43 H5 Z1. M8
G83 G98 R.1 Z-.65 Q.1 F3.
X-1.625 Y-.625
G80 Z1. M9
```

17. Call up T6 and insert a peck drilling canned cycle to counter bore two 7/16 " diameter clearance holes.

Table 8-8 summarizes the code for T6.

A **G83** peck drilling canned cycle can be used for counter boring. In this case, we are using a 7/16" four flute flat bottom end mill to counter bore clearance holes for 1/4–20 SHCS (socket head cap screw) heads.

Table 8-8 Code for T6

```
T6 M6 (7/16 4 Flute EM)
S700 M3
G0 G54 X1.625 Y.625
G43 H6 Z1. M8
G83 G98 R.1 Z-.28 Q.1 F3.
X-1.625 Y-.625
G80 Z1. M9
```

18. Call up T7 and insert a peck drilling canned cycle to drill one .118 " diameter pilot hole.

Table 8-9 summarizes the code for T7.

Table 8-9 Code for T7

```
T7 M6 (#32 Jobber Drill)
S1700 M3
G0 G54 X0 Y.5
G43 H7 Z1. M8
G83 G98 R.1 Z-.65 Q.2 F3.
G80 Z1. M9
```

19. Call up T8 and insert a reaming canned cycle to ream one 1/8 " diameter hole.

Table 8-10 summarizes the code for T8. **G85** begins a reaming/boring canned cycle. **G85** commands the spindle to plunge in and out of a hole at a constant feed rate, which in this case is **F3**. (3 in/min).

Table 8-10 Code for T8

```
T8 M6 (.125 Reamer)
S400 M3
G0 G54 X0 Y.5
G43 H8 Z1. M8
G85 G98 R.1 Z-.56 F3.
G80 Z1. M9
```


20. Call up T9 and insert code to cut the perimeter of the part using a 1/2 " diameter end mill.

Table 8-11 summarizes the code for T9.

There is a lot to explain here, but in essence what this code does is walk a 1/2" end mill around the perimeter of the part, bringing in the overall length and width; it also cuts the 1/4" radii on the corners of the part.This tool path code is the kind where CAM systems and post processors really shine. A post processor could spit out the code for this tool path in a heartbeat, whereas writing the code manually for this tool path is tedious.

Let’s proceed!

G1 commands the spindle to move from point A to point B in a straight line at a commanded feed rate. **G1** is probably the most common G code used in CNC programming.

Table 8-11 Code for T9

```
T9 M6 (1/2 2 Flute EM)
S2000 M3
G0 G54 X-2. Y1.25
G43 H9 Z1. M8
Z.1
G1 Z-.55 F60.
G41 D9 X-1.75 Y1.25 F10.
X1.75
G2 X2.25 Y.75 I0 J-.5
G1 Y-.75
G2 X1.75 Y-1.25 I-.5 J0
G1 X-1.75
G2 X-2.25 Y-.75 I0 J.5
G1 Y.75
G2 X-1.75 Y1.25 I.5 J0
G40 G1 X-1.68 Y1.32
G0 Z.1
Z1. M9
G91 G28 Z0
G28 Y0
M30
%
```

G1 must have a feed rate associated with it. In the first block of code in Table 8-11 showing **G1 —G1 Z-.55 F60.** — the end mill moves from **Z.1** to **Z-.55** at **F60.** (60 in/min)

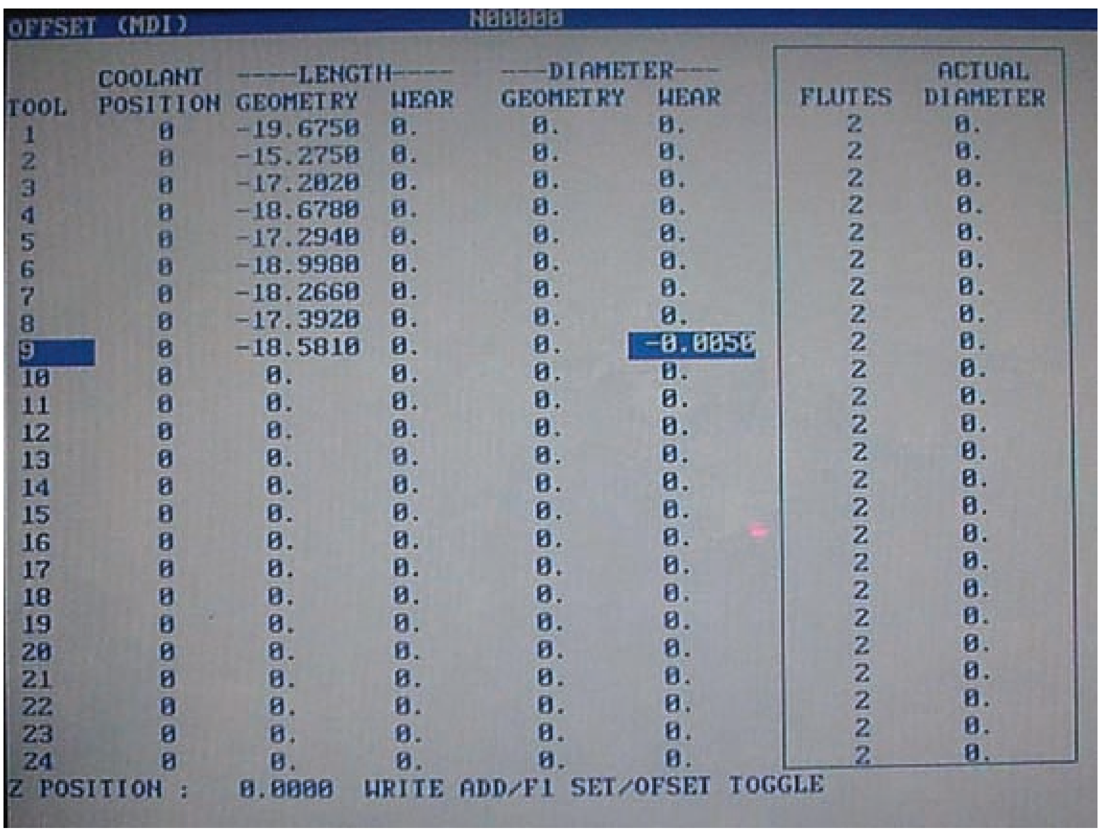
Note that the position of the end mill is well off the part on the X-axis before making this plunge move. You would not want to plunge into material at 60 in/min. or you would likely end up with a broken cutter. The starting position of the end mill was established in a previous line of code: **G0 G54 X-2.5 Y1.25**

G41 is a command that tells the controller to pick up diameter compensation.

D9 tells the controller from which register to pick up the diameter compensation. Once the controller reads G41 in a program, an adjustment to the tool path will be made if you insert a cutter diameter variance into the **D9** "wear" register (see Figure 8-8).

Suppose the end mill diameter was actually .495" instead of .500". Running the part with the tool path values shown in the program would result in the part being .005"oversize in length and width. If you insert a value of -.005" into the **D9** "wear" register, the machine would adjust the tool path and cut the part to the correct size.

Typically you would write a program using a register number that correspond with the tool number being used. In this program we chose the **D9** wear register because it corresponds with Tool 9.


OFFSET (MDI) N00000

TOOL	COOLANT POSITION	----LENGTH---- GEOMETRY	WEAR	---DIAMETER--- GEOMETRY	WEAR	FLUTES	ACTUAL DIAMETER
1	0	-19.6750	0.	0.	0.	2	0.
2	0	-15.2750	0.	0.	0.	2	0.
3	0	-17.2020	0.	0.	0.	2	0.
4	0	-18.6780	0.	0.	0.	2	0.
5	0	-17.2940	0.	0.	0.	2	0.
6	0	-18.9980	0.	0.	0.	2	0.
7	0	-18.2660	0.	0.	0.	2	0.
8	0	-17.3920	0.	0.	0.	2	0.
9	0	-18.5810	0.	0.	-0.0050	2	0.
10	0	0.	0.	0.	0.	2	0.
11	0	0.	0.	0.	0.	2	0.
12	0	0.	0.	0.	0.	2	0.
13	0	0.	0.	0.	0.	2	0.
14	0	0.	0.	0.	0.	2	0.
15	0	0.	0.	0.	0.	2	0.
16	0	0.	0.	0.	0.	2	0.
17	0	0.	0.	0.	0.	2	0.
18	0	0.	0.	0.	0.	2	0.
19	0	0.	0.	0.	0.	2	0.
20	0	0.	0.	0.	0.	2	0.
21	0	0.	0.	0.	0.	2	0.
22	0	0.	0.	0.	0.	2	0.
23	0	0.	0.	0.	0.	2	0.
24	0	0.	0.	0.	0.	2	0.

Z POSITION : 0.0000 WRITE ADD/F1 SET/OFSET TOGGLE

Figure 8-7 This screen shows a -.005 value in the tool diameter wear register for T9. These little adjustments are often used when trying to hold tight tolerances such as when cutting a bearing race pocket or keyway slot.

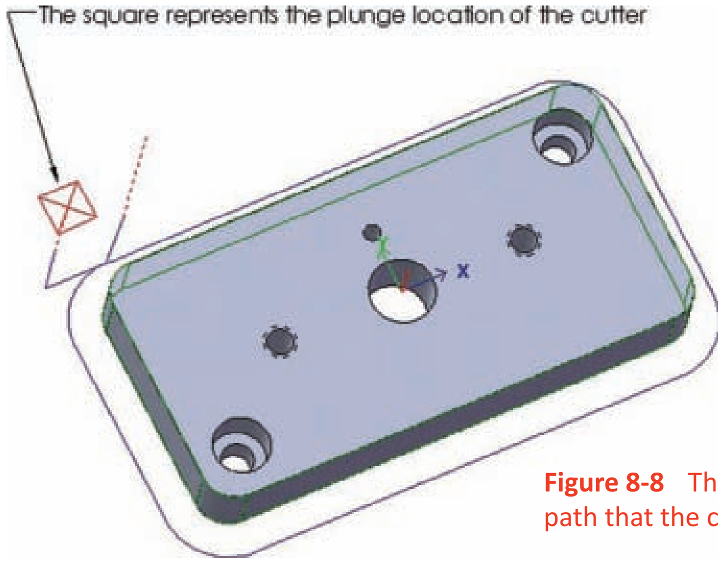


Figure 8-8 The blue line around the part represents the path that the center of the 1/2" end mill would follow.

21. Program with "tool path compensation on," using numbers in a program that represent where the center of the end mill is going to be.

The tool path values shown in this program represent a tool path that is 1/4" outside the part perimeter dimensions; that is, the part was programmed with tool path center compensation. In our CAM system, this is referred to as "tool path compensation on" (see Figure 8-8).

I'm going to interject something here. Programming with tool path center compensation on is the only way I program. I want numbers in a program that represent where the center of the end mill is going to be. It makes dealing with lead on and lead off moves easier. Lead on and lead off moves are important — they are key to avoiding crashes.

If you program with tool path center compensation on, as we are doing in this program, the actual cutter path will follow the values shown in the program either precisely or very nearly, depending on the value placed in the "wear" register. Typically these wear values are small and usually represent the amount an end mill is undersize. When you program with tool path center compensation on, these compensation commands are very easy and intuitive to use. You simply enter the amount an end mill is undersize into the appropriate wear register and away you go.

Compensation commands are useful when using reground end mills. Inserting the amount an end mill is undersize into a wear register will compensate for the undersize diameter and cut the part to the correct size. As a side note, many new end mills vary from their nominal diameters by as much as a few thousandths. This is often an aggravating thing to deal with when trying to hold tight tolerances.

You probably wouldn't worry about this adapter plate being a little oversize because the perimeter of the part would likely fit nothing but "air." However, if you are making something that needs to fit precisely then these compensation commands become very useful.

If there is no compensation value placed in a wear register, then the controller runs a G41 command as if it were looking at a G1 command, which is just a straight move.

Because G41 is a cutting move, a cutting feed rate of F10. is placed in the first G41 line of code. In this case, we went from a feed rate of F60. for plunging the tool at a location which was off the part to F10. for cutting the perimeter of the part.

There is some "peach fuzz" associated with using these compensation commands that can get a little confusing. I'll try to sort this out as painlessly as possible.

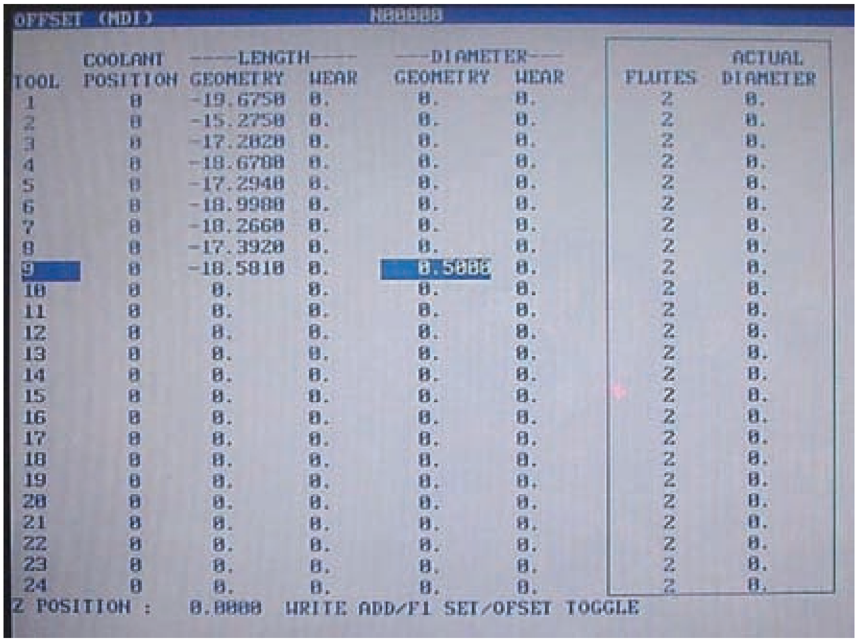
G41 is defined as "cutter compensation left" and **G42**, which is not used in this program, is defined as "cutter compensation right." One of the confusing things about these commands is that both of them can move the cutter either left or right, depending on whether a negative or positive value is placed in a wear register. If no value is placed in a wear register, you could actually substitute **G42** in place of **G41** in this program and the cutter path would be identical. Table 8-12 summarizes how the value of **G41** and **G42** affects the direction they move the cutter.

This begs the question, "Why are there two compensation commands if they both can move the cutter either way?"

The reason is that not all programming is done with tool path center compensation on. You could program the part with tool path center compensation off, that is, you could program the part using numbers in the program that are the same as the outside dimensions of the part, unlike this example.

Table 8-12 Comparing G41 and G42

Code	Value	Direction
G41	Positive	Moves cutter left
G41	Negative	Moves cutter right
G42	Positive	Moves cutter right
G42	Negative	Moves cutter left



OFFSET (MDI)		N00000					
	COOLANT	LENGTH		DIAMETER			ACTUAL
TOOL	POSITION	GEOMETRY	WEAR	GEOMETRY	WEAR	FLUTES	DIAMETER
1	0	-19.6750	0.	0.	0.	2	0.
2	0	-15.2750	0.	0.	0.	2	0.
3	0	-17.2020	0.	0.	0.	2	0.
4	0	-10.6700	0.	0.	0.	2	0.
5	0	-17.2940	0.	0.	0.	2	0.
6	0	-10.9900	0.	0.	0.	2	0.
7	0	-10.2660	0.	0.	0.	2	0.
8	0	-17.3920	0.	0.	0.	2	0.
9	0	-10.5810	0.	0.5000	0.	2	0.
10	0	0.	0.	0.	0.	2	0.
11	0	0.	0.	0.	0.	2	0.
12	0	0.	0.	0.	0.	2	0.
13	0	0.	0.	0.	0.	2	0.
14	0	0.	0.	0.	0.	2	0.
15	0	0.	0.	0.	0.	2	0.
16	0	0.	0.	0.	0.	2	0.
17	0	0.	0.	0.	0.	2	0.
18	0	0.	0.	0.	0.	2	0.
19	0	0.	0.	0.	0.	2	0.
20	0	0.	0.	0.	0.	2	0.
21	0	0.	0.	0.	0.	2	0.
22	0	0.	0.	0.	0.	2	0.
23	0	0.	0.	0.	0.	2	0.
24	0	0.	0.	0.	0.	2	0.

Z POSITION : 0.0000 WRITE ADD/F1 SET/OFSET TOGGLE

Figure 8-9 If you program a part with cutter compensation off, you have to enter the end mill diameter in the geometry register.

If you were to program this part with tool path center compensation off, which is to say with numbers that are the same as the outside dimension of the part, then you would have to enter the end mill diameter in the corresponding "geometry" register (see Figure 8-9). In this case, the end mill diameter is **.500**", which is a positive value.

Once the controller reads **G41** in the program and picks up the geometry of the end mill, the machine adjusts .250" left of the programmed tool path. With G42 the machine adjusts .250" right of the programmed tool path. This makes the definitions of G41 (cutter compensation left) and **G42** (cutter compensation right) understandable.

Once the controller reads **D9** in the program, the controller will pick up both the geometry value and the wear value, and adjust accordingly.

I don't want to beat a dead horse here, but this type of programming with tool path compensation off is not my cup of tea. The adjustments the machine has to make to get the tool in the correct position are relatively large. In this case, the machine has to adjust .250" to get the tool to the correct position to cut the perimeter of the part.

These large adjustments can get you into trouble if you are not completely sure of what path the end mill is going to take to get to the correct position. You may end up cutting something you did not intend to cut.

Let's move on to discuss cutting arcs.

22. G2, which is used for cutting an arc, is defined as circular interpolation clockwise. (see Fig. 8-10)

This means the tool is going to move clockwise in an arc. **G3**, which is not used in this program moves the tool counter-clockwise.

The X and Y values shown in the first **G2** line of code — **G2 X2.25 Y.75 I0 J-.5 —** represent the position of the end mill at the end of the arc. The starting point was set in previous lines of code and is at **X1.75 Y 1.25.** The I and J values which define the size of the arc are input based on the end mill position at the end of the arc.

I and J values can be a little confusing. They are actually quite simple once you understand them. Think of I and J values in terms of "how you would get back to the center of the radius from the I and J values." This is a little unscientific but it works for me. Again, the I and J values must be input based on the position of the end mill at the end of the arc.

With that in mind, the I value, which corresponds with Y, is zero because it is in line in the Y direction with the center point of the radius (see Figure 8-10). The J value corresponds with X is -.500 because the center point of the radius is 1/2" in the negative direction from the J value. If you think of I and J values as "how you would get back to the center of the radius from the end of the move," you can't go wrong.

Figure 8-10 shows how I and J values are derived.

Note that I and J values are something that I rarely mess with as these values are spit out painlessly with a CAM system.

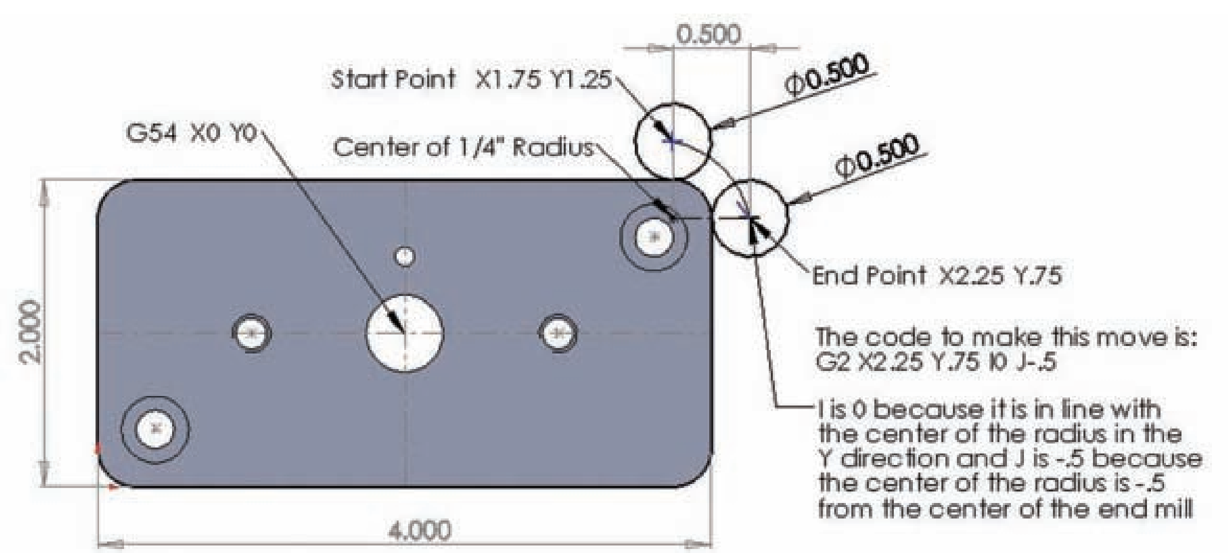


Figure 8-10 How I and J values are derived.

23. G40 cancels circular interpolation. The remaining commands are used to clear the cutter and end the program.

After the arcs are cut, the next step is a lead off move, which is called out in the line: **G40 G1 X-1.680 Y1.320**. This move clears the cutter slightly away from the perimeter of the part so when the cutter retracts, it does not rub against the part.

The next couple of moves retract the end mill slightly above the part at “rapid” speed: **G0 Z.1** followed by **Z1. M9. M9** turns the coolant off.

G91 G28 Z0 retracts the spindle to “machine zero” or the home position in **Z**, where tool changes take place.

The next line: **G28 Y0** moves the machine table to the home position or “machine zero” in the **Y** direction, which is as close to the operator as the table will go. This position makes changing parts in the vise easy for the operator.

M30 signifies the end of the program and tells the controller to rewind to the beginning of the program so another part can be loaded and run.

The percent sign (%) at the end of the program means the game is over.

24. Delrin® is a wonderful material to machine. (see Fig. 8-11).

The run time for this program was 6 minutes, 33 seconds. Being that the material is Delrin® plastic, which is soft yet rigid, the run time could have been reduced substantially by increasing feeds and speeds.

As a side note, I’ve found that run times become an issue when there are a large number of parts to complete. With plastic and aluminum parts, run times can be reduced simply by increasing feeds and speeds at the controller. If you double the feed rate and spindle RPM for a specific cutter, the chip load remains the same, but run time is cut in half. With tougher materials like stainless steel, it’s not so easy. You may be able to increase feed rates, but you’ll soon run into trouble increasing spindle RPM beyond a cutter’s recommended surface feet per minute (SFM) for a specific material. The cutter will wear prematurely.

Depth of cut (DOC) also has a lot to do with run time and is something I struggle with. I frequently find myself going back to the computer to increase depth of cut once I see a part run. Too much DOC puts a lot of stress on cutters whereas too little substantially increases run time. Depth of cut is not something you can easily change at the controller. That input has to be made at the computer if you are programming with a CAM application.

Figure 8-11 This is the adapter plate after all the tools have been run. The only thing left to do is face off the back side and bring the part to the correct thickness.

25. Become familiar with these commands so that you can mill a substantial number of parts that come through in a typical shop. (see Fig. 8-12).

An additional command that could be useful in this program is the dwell command. The dwell command can be called out in a canned cycle as the letter P followed by time in seconds. For example, we could have used the dwell command in the canned cycle to counter bore the two 7/16" diameter clearance holes as follows: **G83 G98 R.1 Q.1 Z-.280 P.5**

The dwell command is not something I use much, but it can be useful for producing a clean bottom cut on a counter bored or counter sunk hole.

With the handful of commands shown in the programming example above, many useful parts can be milled. Figure 8-12 shows a small sampling of parts machined in our shop using nothing but the commands introduced so far.

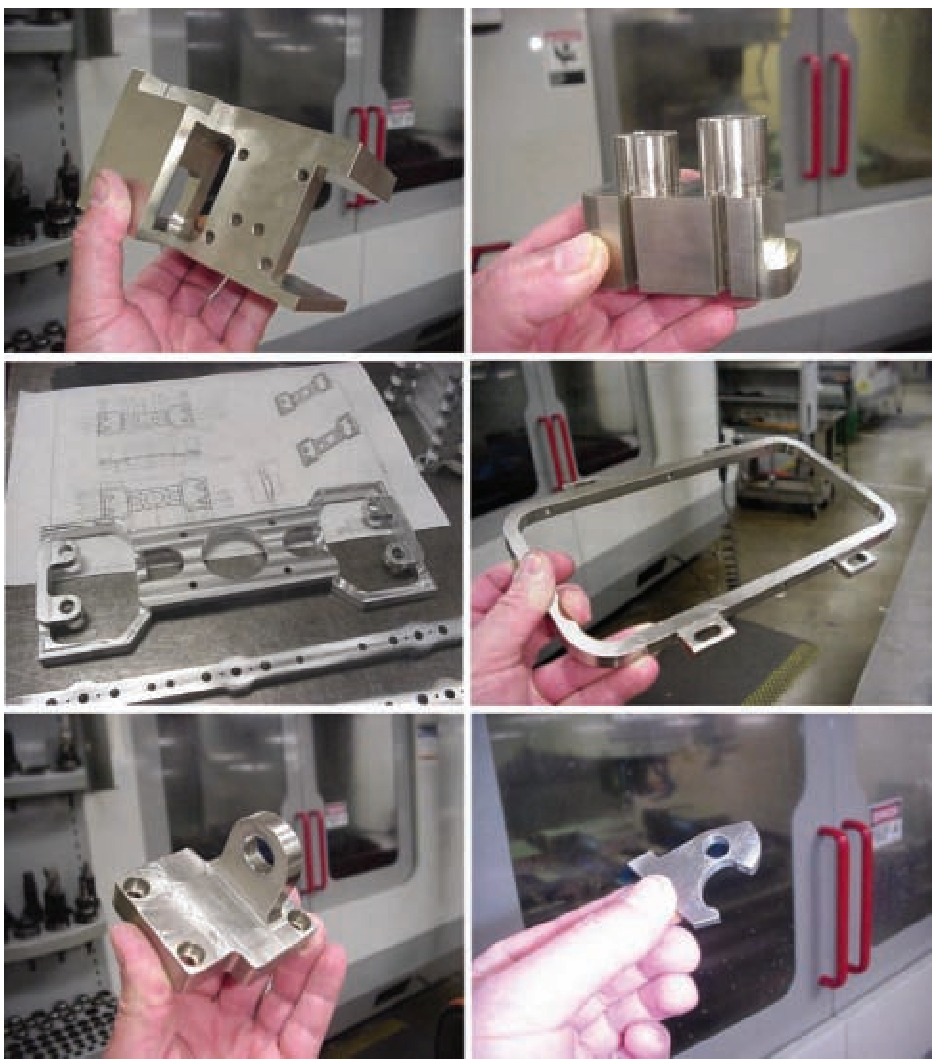


Figure 8-12 These parts were all made using nothing more than the commands described thus far. The programs to make these parts, however, were generated using CAM software.

26. Once you have completed the programming for setup #1, begin the programming for setup #2. (see Figs. 8-13 and 8-14)

In the example we have followed thus far for the adapter plate, the first setup (O00101) involved a variety of tools to cut the part. Here, the purpose of the second setup (O00102) is to mill the part to the correct thickness.

There is approximately .170" of material on the backside of this part to be machined off. Touch off the 3" diameter cutter to a 1/2" gauge block placed on top of the parallels you are going to use and set the tool length offset for T1 (see Fig. 8-14).

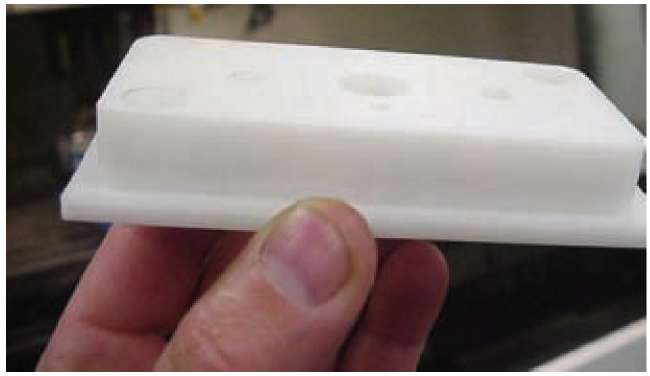


Figure 8-13 The adapter plate with excess material still on the back side.

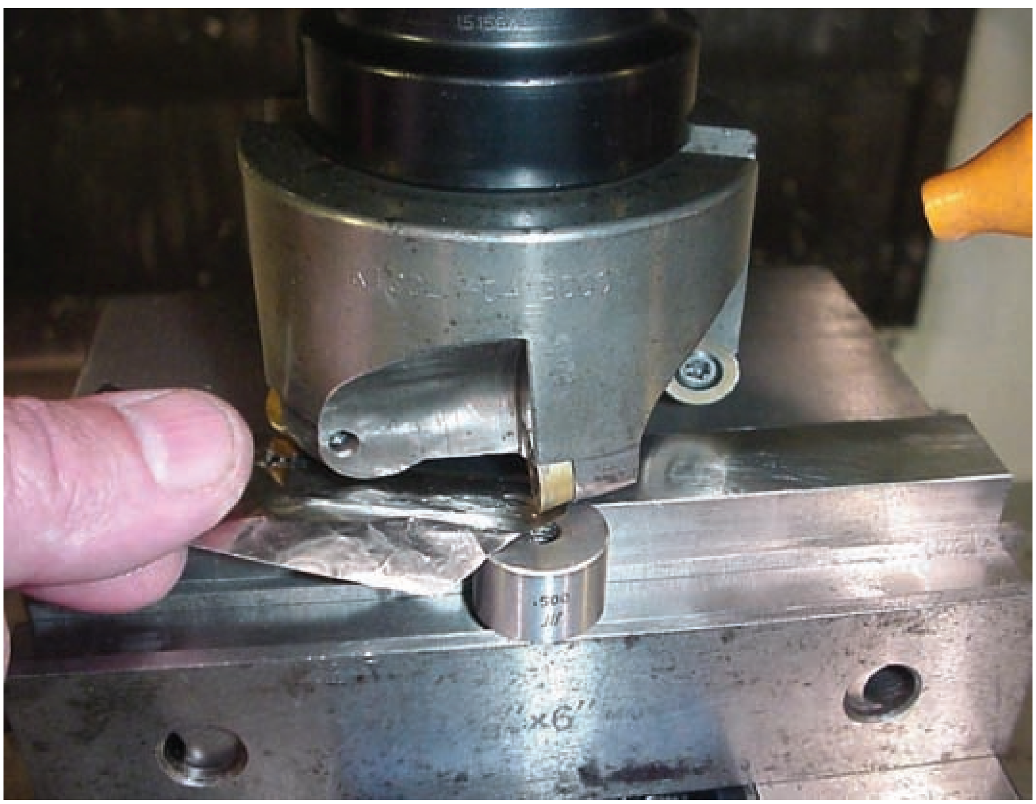


Figure 8-14 Getting ready to face off the back side of the adapter plate by setting the facing tool 1/2"above the parallels. The Z position of this facing tool will then be input into the tool length offset register.

27. Set the G54 starting point to approximately the center of the block. (see Fig. 8-15)

The G54 starting point is set approximately to the center of the block so the numbers in the program are easy to follow. Again, the starting position does not have to be too accurate for this operation because a 3" diameter cutter is going to be used to face off a 2" wide part.

28. Complete setup #2.

All the commands used in the program for setup #2 have been described previously. Table 8-13 summarizes this program.

As you can see, the depth of cut (DOC) is .050" to whack off the excess material with successive passes. The final cut amount is .005". All Z values in the program are positive values except the final cut. The final .005" cut at Z0, highlighted in Table 8-13, will bring the part to the correct thickness.

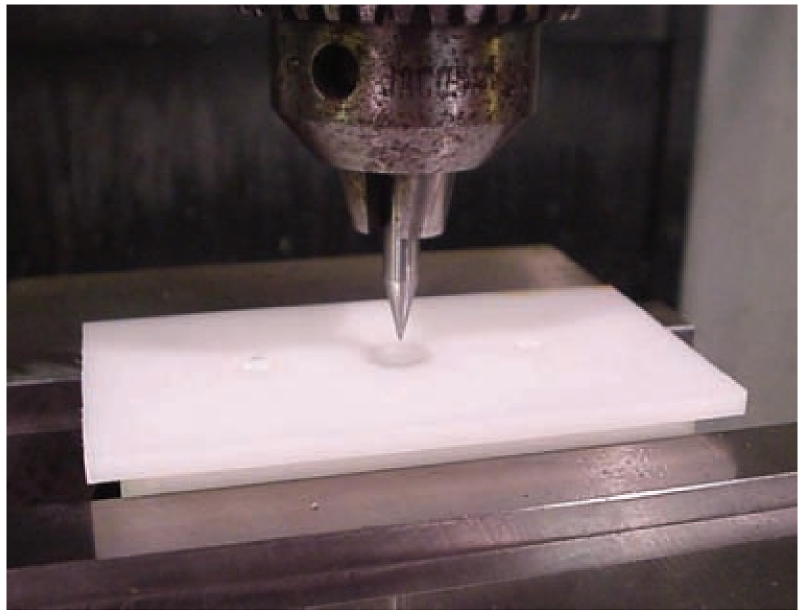


Figure 8-15 The G54 starting point for the facing program is being selected by using a pointer. The X and Y positions are not critical here because the facing tool is much wider than the part.

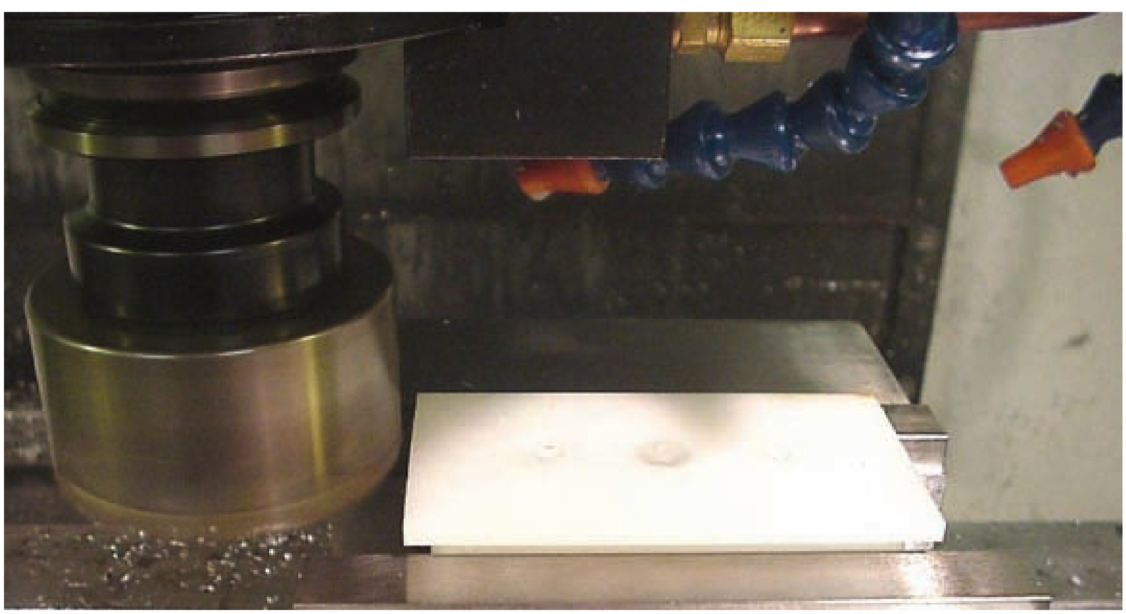


Figure 8-16 In many instances, it is critical that cutters do not plunge down on material. Always start a cutter off the material when you can.

Table 8-13 Code for Setup #2

```
%
O00102 (Adapter Plate Setup #2)
G17 G20 G40 G49 G54 G64 G80 G90 G98
T1 M6 (3" 4 Flute EM)
S2000 M3
G0 G54 X-3.75 Y0
G43 H1 Z1. M8
Z.3
G1 Z.12 F60.
X3.75 F30.
G0 Z.3
X-3.75
G1 Z.07 F60.
X3.75 F30.
G0 Z.3
X-3.75
G1 Z.02 F60.
X3.75 F30.
G0 Z.3
X-3.75
G1 Z.005 F60.
X3.75 F30.
G0 Z.3
X-3.75
G1 Z0 F60.
X3.75 F30.
G0 Z.3
Z1. M9
G91 G28 Z0
G28 Y0
M30
%
```

29. Be certain you have adequate room for a lead on move. (see Fig. 8-16)

You don't want the cutter to plunge down into uncut material as it gets ready to take the lateral cutting moves. The position is X-3.75 Y0. The lead off move is the same distance from the starting point in the positive direction. This will insure that the entire diameter of the cutter passes across the length of the part for a smooth finish.

Run time for the program is 1 minute, 34 seconds. Because the material is soft, you could probably whack off the material with fewer passes. The "depth of cut" for roughing off material always seems to pose interesting dilemmas. In general, sturdy end mills and sturdy setups allow you to increase depth of cut.

The clearance plane is set to Z.3 in this example. This value is important because the usual default clearance plane of Z.1 would not be high enough to clear the cutter over the excess material on a return move and a crash would occur. I'm familiar with this mistake…I've done it about three too many times.

As you can see, programming this part is not difficult. What is difficult is making sure everything is correct. The finished adapter plate is shown in Figure 8-17.

There are other ways this part could have been made; in fact, there are infinite numbers of ways it could have been made. But this is how I would do it.

We're now going to move on to a brief discussion of three-dimensional or 3D cutting.

Figure 8-17 The completed adapter plate.

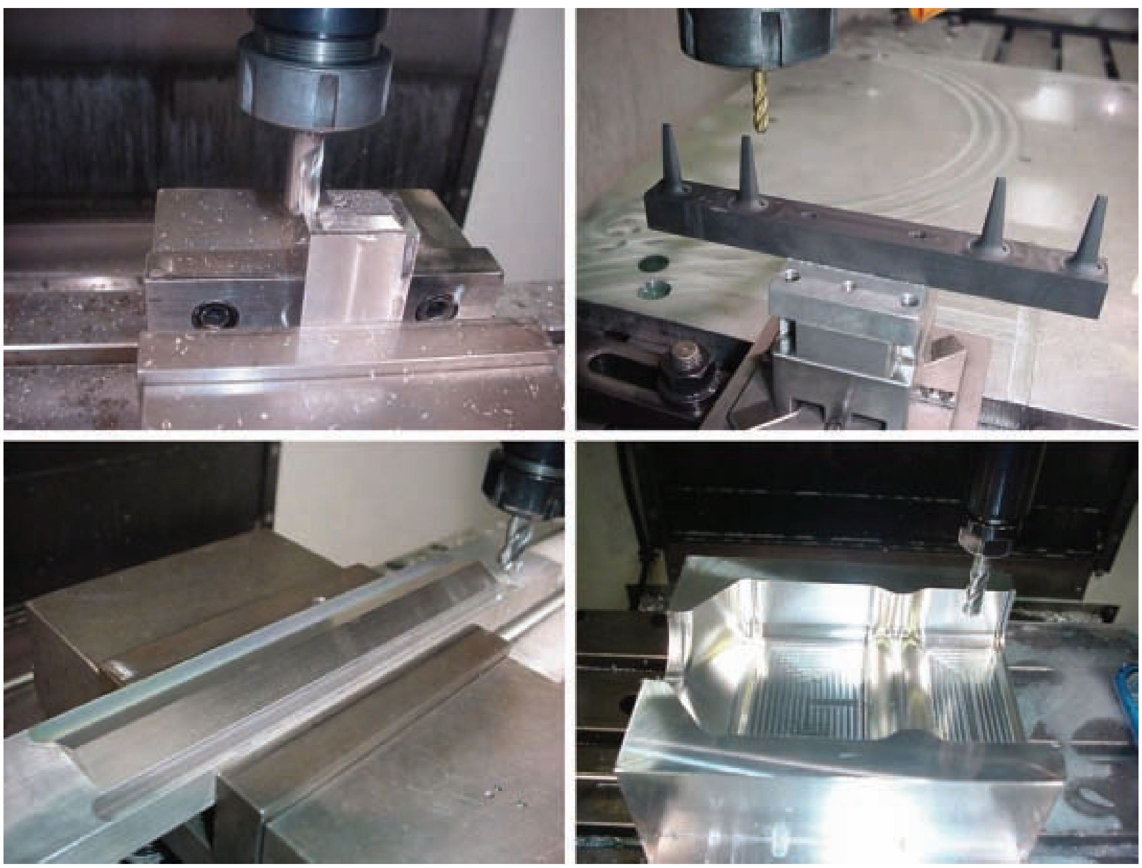


Figure 8-18 Many of the surfaces on these parts were cut with 3D moves.

30. 3D cutting enables you to cut a variety of non-flat surfaces in one setup. (see Figs. 8-18 and 8-19)

Reducing setups is usually a great time saver. Reducing setups can also increase part accuracy because there is less tolerance used up due to inaccuracies in vise clamping and edge finding.

3D surfaces can be anything from simple angle cuts to complex surfaces. The vast majority of injection molded consumer products have 3D surfaces.

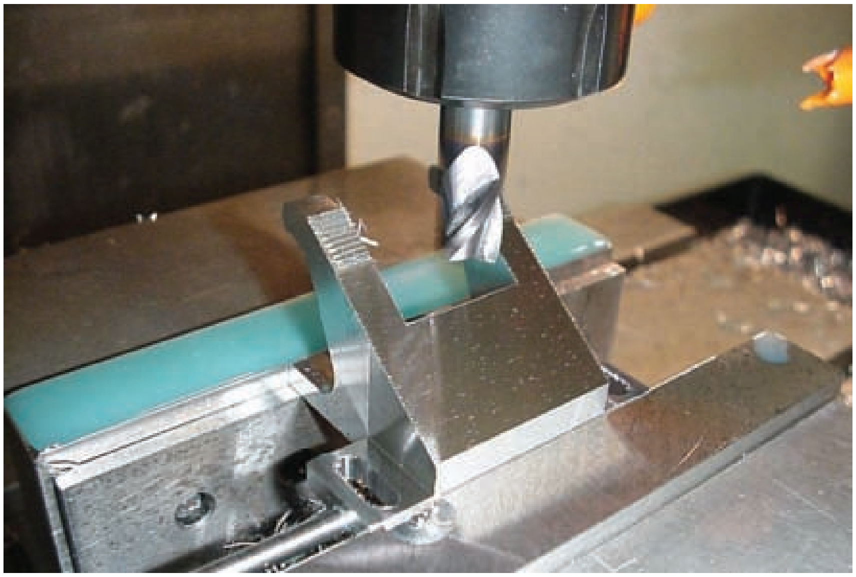


Figure 8-19
This example shows an angle being finished using 3D cuts. The angle was roughed in using a roughing tool. A portion of the roughed in surface can be seen near the top of the part.

When I first saw how CNC machines could cut 3D surfaces, I was pretty amazed. People just getting into machining may never appreciate what old timers had to go through to create smooth, curved surfaces. There was a lot of craftsmanship and handwork involved. These relatively new CNC machines in combination with the software available can produce precise 3D surfaces effortlessly. Sometimes it feels like I'm cheating.

Figure 8-19 shows an example of an angle cut being made with 3D programming.

3D programming to cut an angle is basically programming a cutter to move up and over a certain amount, taking a lateral cut, then repeating the process

> As fast as CNC machines are at making parts, I've found that customer's expectations of how fast something can be made always seems to stay a little ahead of reality.
>
> One restless customer I recently dealt with asked me, "Don't you just stick the stock in the machine, press a few buttons and out pops the part?" My answer was "No."

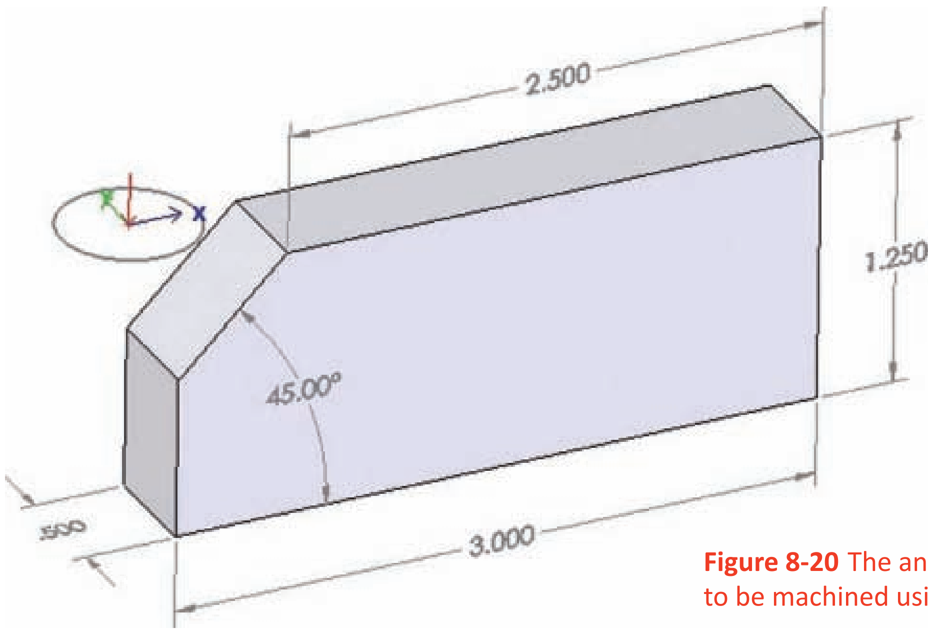


Figure 8-20 The angle on this part is going to be machined using 3D cuts.

31. You can complete 3D cutting on a conventional milling machine. (see Fig. 8-20)

In the early 1990s, I worked at a mold shop that had no CNC equipment. Occasionally we'd 3D mill an angle or radius on a part by calculating how far to move the cutter up and over for each successive pass. It was slow and tedious, but it worked.

CNC machines do the same thing. The difference is that you can now let the CAM software do the calculating and hang out somewhere while the cut is being made.

Let's consider how to program the angle cut on the part shown in Figure 8-20.

32. Choose cutter diameters for the program. (see Figs. 8-21)

For this cut, a 1/2" diameter end mill is a good choice. They are usually readily available and are sufficiently sturdy for this cut regardless of the material being cut within reason.

For roughing cuts, I generally choose end mills that are not in great condition. I try to save new, sharp, and specialty cutters for finishing cuts. These choices can be made when setting up at the machine. It's easy to find "thrasher" end mills for your roughing cuts. Sometimes it's not so easy to find sharp end mills suited for finishing work, at least in our shop.

The end mill used for roughing in this example is a sharp cornered, flat bottom, 1/2" diameter end mill. The roughing cuts are programmed to stay away from the finished surface about .010". The step over moves are .05" in the program so the numbers are easy to follow. The roughing cutter path and cuts are shown in Figure 8-21.

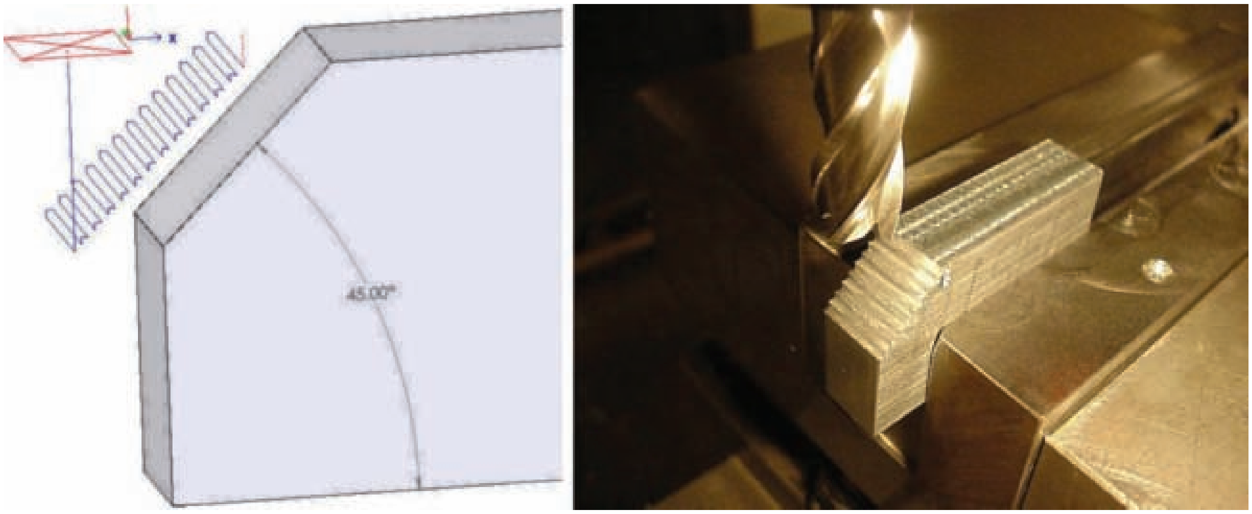


Figure 8-21 The cutter path for roughing this angle is shown on the left and the actual cuts are shown on the right.

33. Hog nose end mills with a small tip radius are often used for finishing 3D surfaces, especially for tough materials. (see Figs. 8-22 and 8-23).

The finishing end mill is a 1/2" diameter hog nose end mill with a .020" radius. Any size radius on the end mill could be used however. We normally buy hog nose end mills with .02 radii for 3D cutting. We chose this radius to standardize, so when we open the drawer and pull out a hog nose end mill we know it will have an .02R.

Figure 8-22 Hog nose end mills cut with less pressure than ball nose end mills.

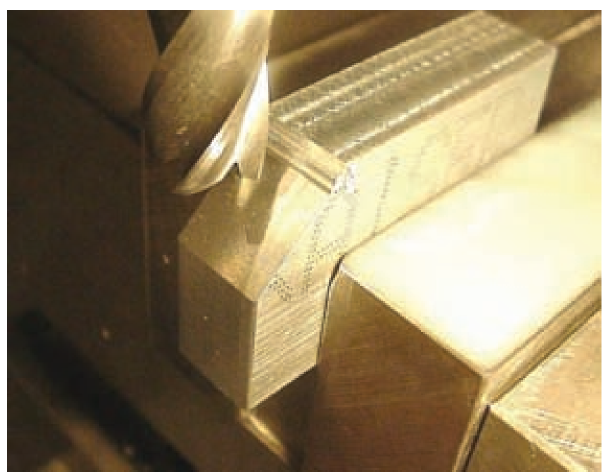


Figure 8-23 A hog nose end mill is shown finishing the angle using 3D cuts.

One benefit of using an end mill with a relatively small tip radius such as .02R is that they cut with less pressure than a ball nose end mill. A ball nose end mill would have more cutting surface in contact with the part, and thus increased cutting pressure, which could lead to chatter and a poor surface finish.

The back left corner was chosen for the **G54** starting point because the corner is easy to edge find and makes following the numbers in the program easier.

34. Write the program.

Writing a program manually to make this 3D angle cut is feasible for the first roughing tool only. The sample program in Table 8-14 illustrates the rough moves for cutting this 3D angle. Normally I would never write anything like this manually. I would let the CAM software spit out the code for me.

The program shows the G-code for the roughing cuts which were written manually to show you what is happening while 3D cutting. The step over moves in this program are .05" up and .05" over. Only a portion of the code for T2, which programs the hog nose finishing end mill are shown because there are far too many finish moves. The code for the T2 finishing hog nose end mill was output using CAM software.

The step over moves for the finishing cuts made by T2 are .004". The larger the step over moves, the coarser the finish and vice versa. We usually program the step over moves for finishing cuts between .002" and .006" when using a hog nose end mill with an .02R, depending on how smooth the finish needs to be.

One may wonder why I chose such a simple example to illustrate 3D programming and not something more exotic like a Coke® bottle. The reason is because the programming logic is the same. In both cases, cutters need to be programmed to move in X, Y, and Z to whittle out a shape.

As you can see, there a lot of decisions that must be made when programming and machining. The primary decisions to be made are: what speeds and feeds are going to be used, what depth of cut, step over amount and cutter sizes, cutting sequence, and how the part will be held. These decisions have to be made taking into account tolerances and part rigidity.

Understand that CAM systems, as good as they are, will not make many of these decisions for you. The decisions are ultimately up to the programmer and machinist.

One thing about using CNC machines is that they are so efficient; you can probably err on the side of conservative cutting parameters and still make good time. In fact, you may make better time in the long run if you are running tough material by virtue of not having to change broken or dull cutters too often.

There is no absolute right or wrong when planning and determining settings. There is a lot of gray area and wiggle room. Many settings are chosen based on feel and experience. Many settings are chosen based on comfort level. Bear in mind that I'm not right all the time. Nobody is.

As you come across some of the examples and tips in this book you may say to yourself…"why didn't he just blah, blah, blah? It would have been so much faster and easier." You may be right.

Table 8-14 Cutting a 3D Angle

```
%
O00101 (3D 45 DEGREE ANGLE CUT)
G17 G20 G40 G49 G54 G64 G80 G90 G98
T01 M06 (1/2 4 FLUTE EM)
S2500 M03
G00 G54 X-.260 Y-.5
G43 H01 Z1. M08
Z.1
G01 Z-.5 F60.
Y0 F25.
Z-.45
X-.21
Y-.5
Z-.4
X-.160
Y0
Z-.4
X-.11
Y-.5
Z-.350
X-.06
Y0
Z-.3
X-.01
Y-.5
Z-.25
X.04
Y0
Z-.2
X.09
Y-.5
Z-.15
X.14
Y0
Z-.1
X.19
Y-.5
Z-.05
X.24
Y0
G00 Z.1 M09
M01
T02 M06 (1/2 HGN EM .020R)
S4000 M03
G00 G54 X-.2441 Y-.5
G43 H02 Z.1 M08
G01 Z-.5058 F35.
Y0 F40.
Z-.5018
X-.2401
Y-.5
Z-.4978
X-.2361
Y0
Z-.4938
X-.2321
Y-.5
Z-.4898
X-.2281
```

And so on…

```
G00 Z.1
Z1. M9
G91 G28 Z0
G28 Y0
M30
%
```

Suggestions for Becoming Familiar with Code

1. The majority of machining operations can be undertaken with just a handful of G- and M-codes.
2. Writing programs manually is a good way to become familiar with G-code.
3. Begin and end all programs with a percent sign.
4. Insert a program number and follow it with a description of the part and setup number in parentheses.
5. Insert safety line coding.
6. Call up a tool number, insert a tool change command, and follow in parentheses with a description of the tool.
7. Insert a spindle speed and turn the spindle on.
8. Insert a "rapid" command and a starting point command; then tell the spindle where to go in X and Y.
9. Call up the tool length offset for T1, move the spindle in Z to the rapid plane, and turn on the coolant.

10. Insert a drilling canned cycle to center drill hole locations.
11. Insert additional hole locations in X and Y for center drilling.
12. Cancel the canned cycle, return to the "rapid" plane, and turn off the coolant.
13. Once the programming for T1 is complete, proceed with the programming for T2 and any additional tools. Describe each tool in parentheses.
14. Call up T3 and insert a peck drilling canned cycle to drive two .201" diameter ¼-20 pilot holes.
15. Call up T4 and insert a tapping canned cycle to tap two ¼-20 pilot holes.
16. Call up T5 and insert a peck drilling canned cycle to drill two 1/4" diameter holes.
17. Call up T6 and insert a peck drilling canned cycle to counter bore two 7/16" clearance holes.
18. Call up T7 and insert a peck drilling canned cycle to drill one .118" diameter pilot hole.
19. Call up T8 and insert a reaming canned cycle to ream one 1/8" diameter hole.
20. Call up T9 and insert code to cut the perimeter of the part using a 1/2" diameter end mill.

21. Program with "tool path compensation on," using numbers in a program that represent where the center of the end mill is going to be.
22. G2, which is used for cutting an arc, is defined as circular interpolation clockwise.
23. G40 cancels circular interpolation.
24. Delrin® is a wonderful material to machine.
25. Become familiar with these commands so that you can mill a substantial number of parts that come through in a typical shop.
26. Once you have completed the programming for setup #1, begin the programming for setup #2.
27. Set the G54 starting point to approximately the center of the block.
28. Complete setup #2.
29. Be certain you have adequate room for a lead on move.
30. 3D cutting enables you to cut a variety of non-flat surfaces in one setup.
31. You can complete 3D cutting on a conventional milling machine.
32. Choose cutter diameters for the program.
33. Hog nose end mills with a small tip radius are often used for finishing 3D surfaces, especially for tough materials.
34. Write the program.

CHAPTER 9

Fire Up a Machine

I started my CNC training doing tool changes in our new mill. That first step was important. Just the act of doing a tool change let me interact with the machine and controller in a way that gave me confidence and let me know that the machine wasn't out to get me. In a weird way, the machine didn't seem to know or care I was a rank beginner.

I knew right away that I was going to like the relationship. The movement of the table, spindle, and tool changer combined with the processing power of the controller was enticing. I wanted to learn more.

1. Write a simple program to practice working with a CNC machine.

One of the first programs I concocted was a drilling program for the milling machine. Once the program was running and started drilling holes, I thought to myself, "We have a winner here! If nothing else, this machine is a drilling maniac." I was surprised how easy it was to change feeds and speeds, and how quickly the machine moved to different hole locations.

2. CNC machines are easy to set up.

I was actually surprised by how uncomplicated the machine was to set up. There are just a few things the machine needs to know before it can run a program. In a nutshell, the first thing it needs to know is where the starting point or origin of the part is — just like you would need to know if you were machining a part in a conventional mill. The second thing the machine needs to know is where the top of the part is for all the tools.

3. Inputting basic information requires some familiarity with the machine and the controller. (see Fig.9-1)

Table 9-1 summarizes important features of a CNC machine that are highlighted in Figure 9-1, and described further in suggestions 4-14.

Table 9-1 A Guide to Key Features of a CNC Machine

Label	Name of Feature	Suggestion Where Discussed
A	POWER ON	4
B	POWER UP / RESTART	4
C	MDI (DNC)	5
D	ATC FWD	6
E	ATC REV	6
F	ERASE PROG	10
G	EOB	12
H	WRITE / ENTER	12
I	HOME	13
J	CYCLE START	14

4. Press the POWER ON button, wait for the controller to boot up, and then press the POWER UP / RESTART button. (see Fig.9-2)

When we first got our machines, our lead man at that time got stuck on this step. He didn't know he had to press the POWER UP / RESTART button after pressing the power on button; therefore, he couldn't get the machine to do anything. The sad thing is, because of his initial disappointment or lack of ambition, he never touched the machine again.

When you press the POWER UP / RESTART button, the machine will run through a short routine to calibrate itself. Once everything stops moving, you can proceed.

Let's continue by executing some tool changes.

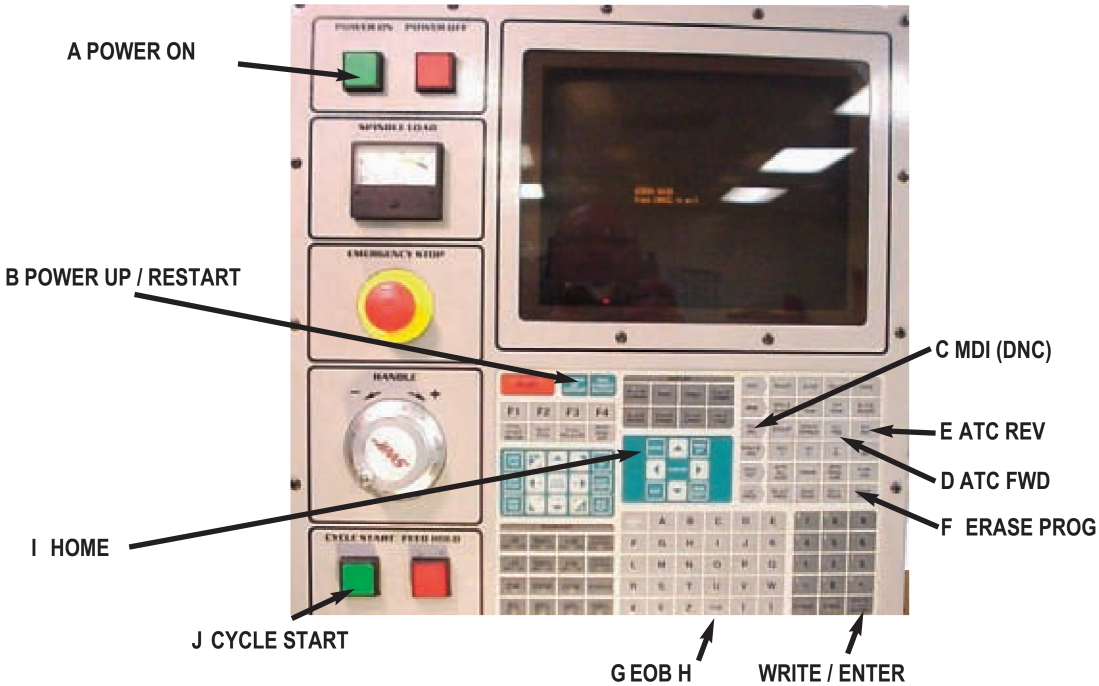


Figure 9-1 A guide to important features on a CNC machine.

Figure 9-2 Pressing the POWER UP / RESTART button runs the machine through a short calibrating routine.

5. Press the MDI button to get into Manual Data Input mode. (see Fig. 9-3)

MDI is the only mode that allows you to change tools manually. When you press the MDI button, the MDI screen comes up. You don't have to enter anything in this screen to make manual tool changes.

6. The ATC FWD and ATC REV buttons will cycle the tool carousel forward and backward, and also actuate the tool-changing arm. (see Fig. 9-3)

Pressing the ATC FWD button in MDI cycles the automatic tool change forward. In turn, pressing the ATC REV button cycles the automatic tool change backward (in reverse). You don't have to have any tool holders in the machine to use these features. Cycle the tool change a few times until you get used to the action and noise. Always keep your hands out of the machine when cycling the tool changer.

7. Call up a specific tool number in MDI mode.

Type T1 or any tool number with the controller keys. Then press either the ATC FWD or the ATC REV button. The tool carousel will cycle T1 into position and the actuating arm will install that tool in the spindle if it is in the carousel. The machine doesn't know if a tool holder is in the carousel or not. If T1 is not in the carousel, you can install one now manually.

Figure 9-3 The MDI , ATC FWD, and ATC REV buttons all help with changing tools.

8. Insert a tool holder in the spindle manually. (see Fig. 9-4)

It's important to line up the spindle lugs with the slots in the tool holder as you press the tool release/engage button on the side of the spindle housing.

Figure 9-4 When installing a tool manually, line up the lugs.

9. Turn the spindle on in MDI.

Press the MDI button to bring up the MDI screen if you are not there already.

10. If any code is on the screen when you go to MDI mode, press the ERASE PROG button. (see Fig.9-3)

When you press this button, a prompt will come up in the screen asking if you want to erase the program. Press Y (yes) on the keyboard to get rid of the code. Don't worry about losing any vital information here. MDI mode is used primarily as a scratchboard for small and temporary programs.

You can now enter a program on the screen. In this case, the program entered will be

S500 M3

The spindle speed is set to five hundred RPM by the S500 command; M3 is the command that turns the spindle on clockwise.

11. M codes are like switches.

Anytime you use an M code in a program, you are generally either switching something on or off, or actuating something such as coolant (M8 or M9), spindle (M3 or M4), tool changer (M6), etc. M codes have other functions as well.

12. Use the EOB button to complete a line of code and the WRITE / ENTER button to enter that line of code into the MDI screen. (see Fig.9-5)

When you are done with a line or a block of a program, press the EOB (end of block) button. Doing so will input a semicolon at the end of that line of code.

S500 M3;

The semicolon is necessary; it tells the controller you are done inputting code on that line or block. The code will show up in the lower left corner of the screen as you type. Note that you don't need to type in any spaces. The controller puts spaces in for you once you "enter" the information.

Pressing the WRITE / ENTER button will enter the typed code into the MDI screen where it can be run.

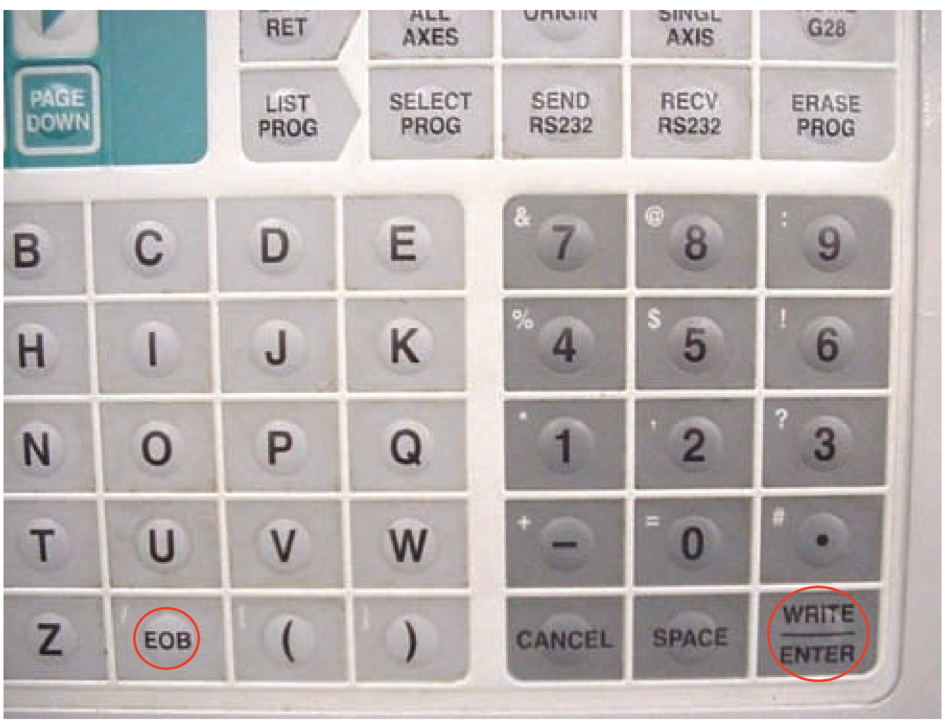


Figure 9-5 This part of the controller shows both the EOB and WRITE/ENTER buttons.

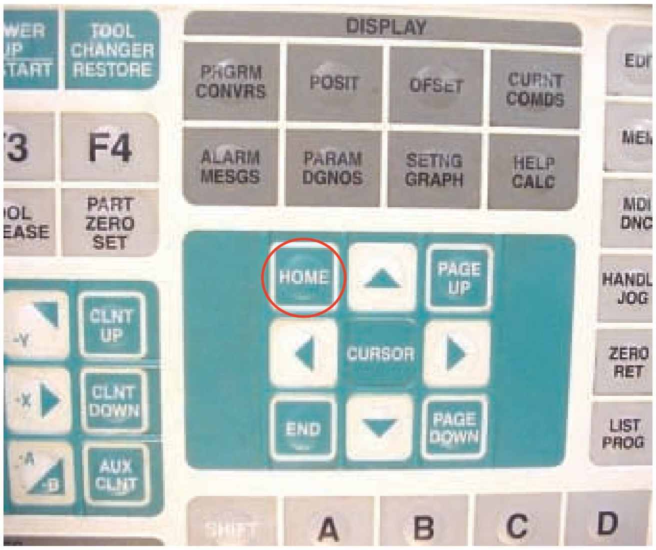


Figure 9-6 The HOME button moves your cursor to the onscreen beginning of your program.

Figure 9-7 When you're ready to run your program, press the CYCLE START button.

13. The HOME button and its neighboring buttons allow you to move around the program while you are writing and correcting code. (see Fig.9-6)

Press the HOME button to return the cursor to the beginning of the program. The other buttons, such as END, PAGE UP, and the arrows, work much as your regular computer keyboard keys work for moving around a document or spreadsheet.

14. Press the green CYCLE START button to run the program. (see Fig.9-7)

If the spindle turns on, which it should, you have successfully written and executed a short program.

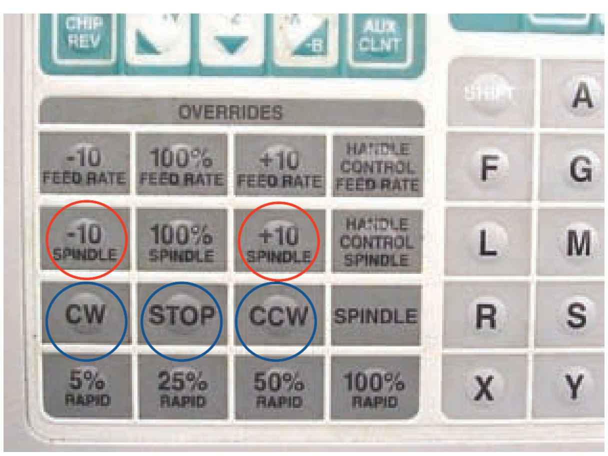


Figure 9-8 The buttons for increasing and decreasing SPINDLE speed are circled in red. The buttons for stop, CW, and CCW are circled in blue.

15. Several additional buttons are helpful as you write your program. (see Fig.9-8)

Press the SPINDLE+ and SPINDLE– buttons to manually change the spindle RPM. Press the STOP button, and the CW (clockwise) and CCW (counter clockwise) buttons to manually stop and start the spindle rotation and direction.

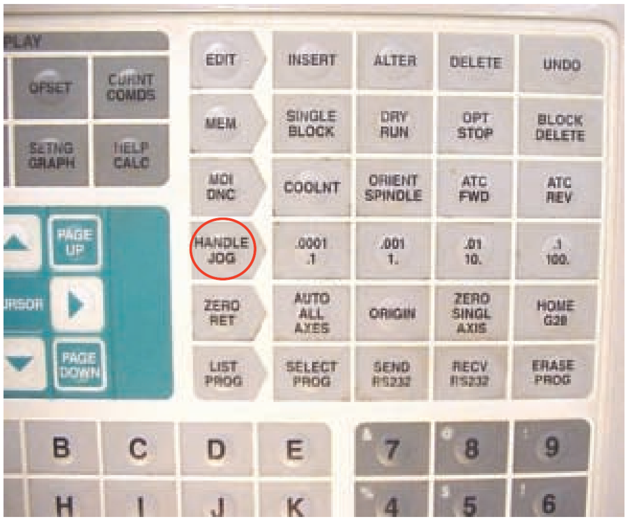


Figure 9-9 The HANDLE JOG button.

Figure 9-10 The main dial for moving the table manually.

16. Use Handle Jog mode to move the table manually. (see Figs. 9-9 through 9-13)

Press the HANDLE JOG button to get into Handle Jog mode (Figure 9-9). Then rotate the hand dial to move the table (Figure 9-10).

Although we know the table is the item that is actually moving during lateral movements, our minds perceive these movements as the spindle head moving. This perception actually works in our favor because it is more in line with how we think about things while indicating, edge finding, and machining.

Therefore, I'm going to let myself off the hook right now to maintain clarity and consistency. I'm going to refer to any lateral movements as the spindle moving. When I say the spindle is moving in an X positive direction, we all know in reality it is not the spindle moving in an X positive direction, but the table moving in an X negative direction. Let's move on.

Once in Handle Jog mode, you can select the axis you want to move along (Figure 9-11), and

Figure 9-11 Once in Handle Jog mode, you can move or cut along an axis of your choice.

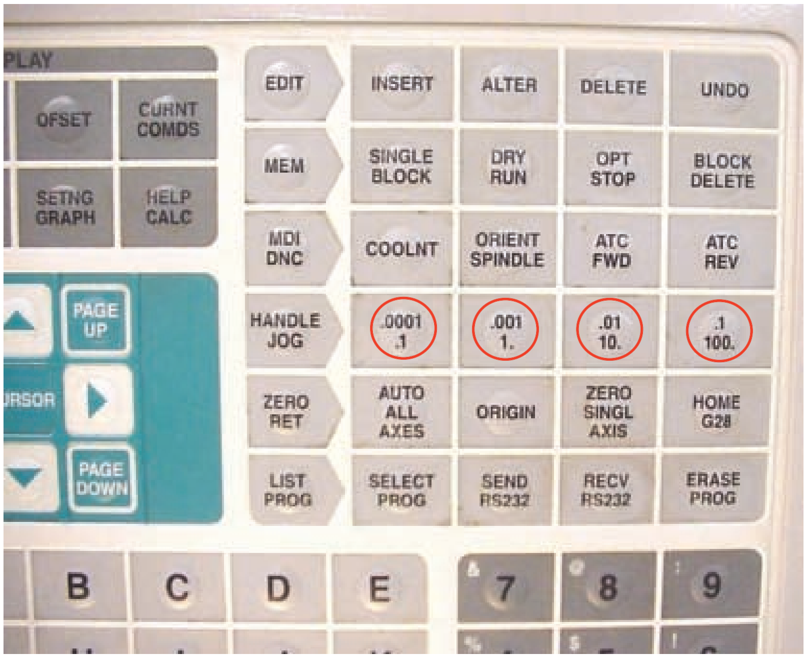


Figure 9-12 Once in Handle Jog mode, you can pick the incremental amount you want the table to move for each click of the handle dial.

choose the increment or "speed" at which you want to move (Figure 9-12).

You have to be careful here. It's been my experience that a lot of crashes are caused by careless "Handle Jogging."

If you push the .01 increment button, then each click of the hand dial will move the spindle/table .010". You have to be especially careful when moving around in .1 per click. Things happen quickly at that setting.

The only time choosing between a plus or minus axis button comes into play is when you want to lock the movement of the spindle or table in a specific direction in Handle Jog mode. You do that with the JOG LOCK button (Figure 9-13).

Figure 9-13 The JOG LOCK button engages the machine to run along the axis of your choice so you don't have to crank the handle dial.

Take a little time to cruise around in Handle Jog mode, changing axes and increments until you get used to how the machine responds. You'll spend a lot of time in this mode indicating and edge finding parts.

I think I'm going to stop here. Though there are many more manipulations you'll need to learn, this introduction gives the reader a feel for becoming familiar with a machine and controller. The manual that comes with your machine will give you all the details you need. Frankly, I dislike reading manuals. I'd rather have a person familiar with a machine go over it with me. A person showing you how to do something can do a lot of filtering for you. With a little practice, manipulating a machine quickly becomes second nature.

Suggestions for Becoming Familiar with a Machine

1. Write a simple program to practice working with a CNC machine.
2. CNC machines are easy to set up.
3. Inputting basic information requires some familiarity with the machine and the controller.
4. Press the POWER ON button, wait for the controller to boot up, and then press the POWER UP / RESTART button.
5. Press the MDI button to get into Manual Data Input mode.
6. The ATC FWD and ATC REV buttons will cycle the tool carousel forward and backward, and also actuate the tool-changing arm.
7. Call up a specific tool number in MDI mode.
8. Insert a tool holder in the spindle manually.
9. Turn the spindle on in MDI.
10. If any code is on the screen when you go to MDI mode, press the ERASE PROG button.
11. M codes are like switches.
12. The EOB and WRITE / ENTER buttons are important tools as you complete your work.
13. The HOME button and its neighboring buttons allow you to move around the program while you are writing and correcting code.
14. Press the green CYCLE START button to run the program.
15. Several additional buttons are helpful as you write your program.
16. Use Handle Jog mode to move the table manually.

CHAPTER 10

Odds and Ends

Tips that shed light on some aspect of machining are tips worth mentioning. The tips presented in this chapter are straightforward and somewhat random; I hope they'll add to your machining knowledge.

A time will come when working in a shop that you'll be obliged to attempt a technique or cutting routine you've never tried before. Those times can be a little unnerving. Most customers and employers want positive results immediately, not leaving much room for error. In this chapter I'll present a few unusual methods of machining that worked for me.

Figure 10-1 Silicone mold release is sprayed on a tool holder for easier tool changes.

Figure 10-2 Scotch Brite ™ is used to remove rust from this tool holder.

1. Use silicone mold release on tool holders for easier tool changes. (see Fig. 10-1).

Some tool changes can be bone jarring. This is especially true when a holder has been in the spindle for awhile. You should never leave a tool holder in the spindle for an extended length of time such as overnight. I've found that using a little silicone mold release on tool holders provides for smoother tool changes.

2. Use Scotch Brite® to remove rust and gunk from tool holders. (see Fig. 10-2).

The task of maintaining acceptable water/oil coolant ratios in machines often gets over looked. In the real world, water just gets poured into machines when coolant levels get too low. Among other things, this practice can lead to rusty tapers on tool holders. An easy way to remove rust from tool holders is to use Scotch Brite®.

3. 3D printing prototype parts can save time (see Fig. 10-3).

Often designers are not completely sure of themselves when designing prototype parts. They'll order one part from the machine shop and, if it works, they'll order more. This method is inefficient from a shop's perspective: Shop personnel have to find material, come up with cutters, program, setup, and run just one part. If the part doesn't work, the process starts

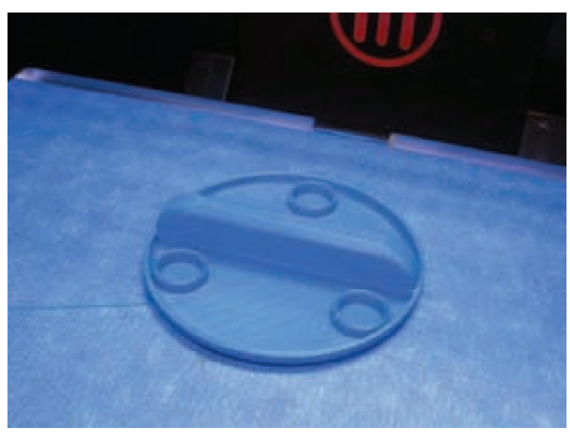


Figure 10-3 3D printers are great tools for making and verifying prototype parts.

over. 3D printers are easy to use. No cutters, stock, or programming are necessary. The only thing needed is a 3D CAD model and a printer, of course.

Once you have a CAD model, all you have to do is print the part, possibly tap some holes, and hand it to the designer. If all is well, the designer will order machined parts only once.

4. Reduce burrs by re-running the intial facing tool over the part after all features have been machined. (see Fig. 10-4).

Burrs are a fact of life in machining. Any cutting tool will throw a burr, sharp tools less so. Burrs must be removed from machined parts by hook or by crook (usually by file). If you re-run the initial facing tool over the part after you have machined features such as holes and the perimeter of the part, you reduce the effort it takes to remove burrs simply because many of them get cut off with this final pass. This is a technique I use often.

5. Cut hard material with carbide, use no coolant, and have patience. (see Fig. 10-5).

After parts get hardened by heat treating, they often need rework for various reasons. Ugh. You can't machine hardened parts like softer materials, but you can still do it. Use carbide, no coolant (just an air stream if available), and relatively slow spindle speeds, and take light depths of cut (.002"–.003").

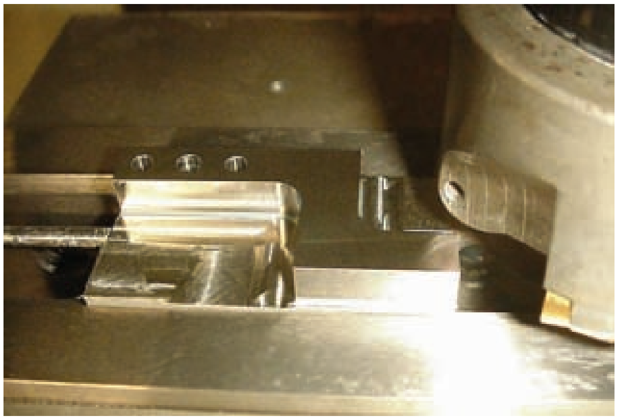


Figure 10-4 A final pass with the initial facing tool helps remove burrs.

Figure 10-5 Light cuts were used to rework this hardened part using a carbide cutter.

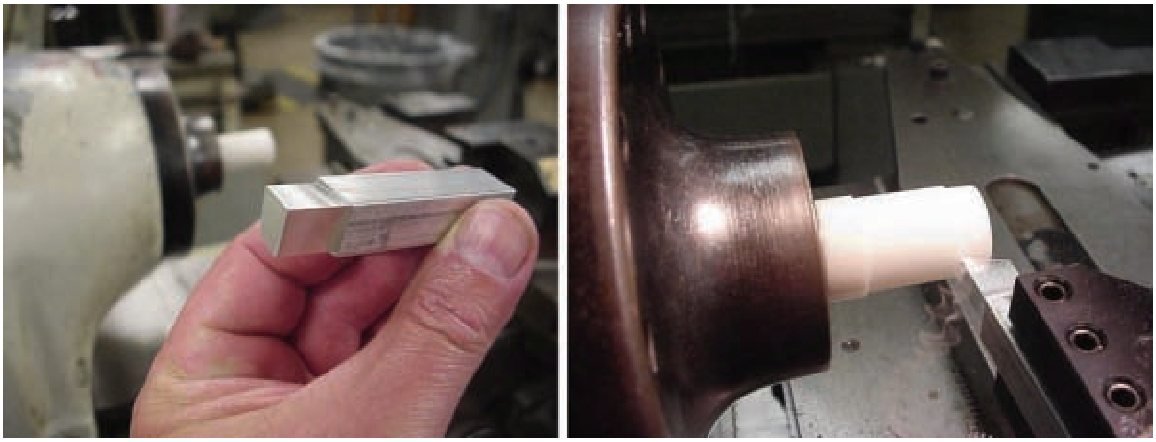


Figure 10-6 A Delrin® part is being machine with an aluminum lathe tool.

6. Cutting tools only have to be harder than the material they are machining. (see Fig. 10-6)

This tip sounds obvious within the realm of machining. I took this concept a step further one day just to experiment. I ground a piece of 7075-T6 aluminum like a lathe tool and proceeded to cut a Delrin® lathe part. Delrin is a rigid plastic which is relatively stable and easy to machine. The aluminum tool held up well and left a nice finish on the Delrin.

7. Use negative rake end mills to push flimsy parts down onto your parallels and holding fixtures. (see Fig. 10-7)

The cutting force of negative rake end mills tends to push the material being cut toward the bottom of the end mill. You can use this force to your advantage when machining parts that are not held too securely. The force tends to push parts against parallels instead of pulling them off, as would positive rake end mills. The rake angle of a cutter is the angle that the chip slides over.

Figure 10-7 A negative rake end mill is being used to machine a long flimsy bar. The downward cutting force helps hold the bar in place.

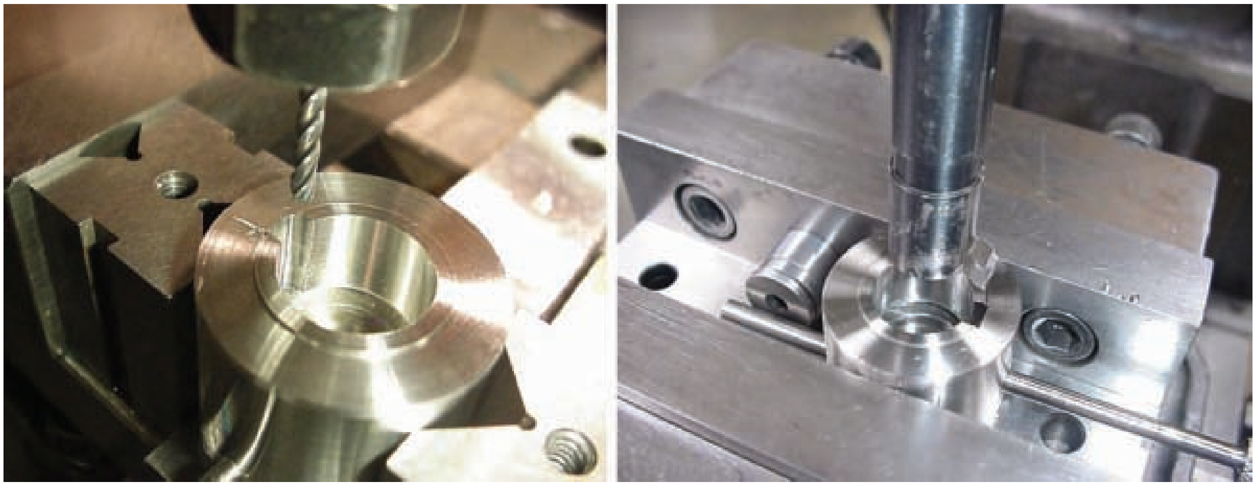


Figure 10-8 CNC machines can be set up to broach. A blind keyway slot would normally be machined by electro-discharge machining (EDM), but the broaching alternative used here made for an interesting job and worked well.

8. Use a CNC machine in a pinch to broach blind keyway slots. (see Fig. 10-8)

Blind keyway slots are features that would normally be cut using electro-discharge machining (EDM). If you don't have an EDM machine, you can still cut blind keyways with a CNC machine. The programming and setup can be tricky because it's probably not something you do often. The spindle must be lined up and locked using an M code. The first thing to do is rough out as much of the keyway as practical with an end mill (left photo of Figure 10-8). Find or make a sturdy broaching tool. Then program the tool to take light cuts (.001") as it enters the roughed out slot. Use cutting oil. I've found this method to be faster than EDM machining.

9. Experiment with unorthodox cutting routines once in a while. (see Fig. 10-9)

Unorthodox machining methods don't always work, so be prepared. One day an order came in for about a dozen teardrop-shaped parts shown in figure 10-9. I asked myself, "How in the heck am I going to hold this part for machining?" After going through various planning scenarios, I came up with an unorthodox plan that would do the entire job in one setup. Co-workers thought I was crazy. I thought I was crazy, but the method worked. I chose to stand a bar of aluminum on end, and cut the profile very close to size with a keyway cutter while stepping down the cutter a little at a time. The bar of aluminum provided the rigidity I needed to make the cuts. As you can see, the finish on the parts after machining was not great, but acceptable for final polishing.

Figure 10-9 Unusual methods of machining sometimes work.

10. When engraving, make sure the surface being engraved is flat. (see Fig. 10-10)

Engraving is a task that CNC machines do splendidly. I remember in one shop where I worked having to use a pantograph to do engraving. What a pain. With CAD/CAM equipment, engraving is a breeze. I like to use "off the shelf" ball nose end mills for engraving when I can. With relatively large lettering and symbols, such as the lettering shown in Figure 10-10, balls nose end mills work well. For small lettering, often seen in mold cavities, pointed tools are commonly used. Because engraving cuts are shallow, you have to make sure the surface being engraved has been either indicated flat or cut flat (unless you are engraving over a 3D surface). Uneven engraving stands out like a sore thumb, especially when using ball nose cutters. With flat bottom cutters, you can get away with more unevenness across the surface being engraved.

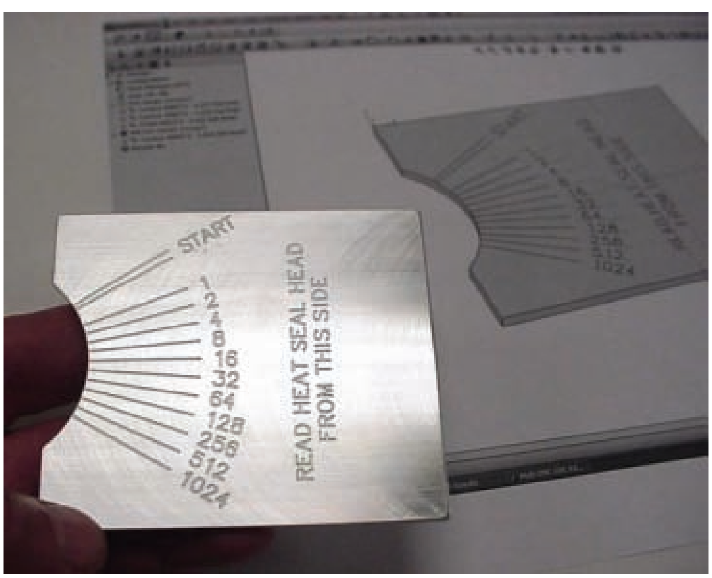


Figure 10-10 For light engraving, the surface being engraved must be flat for consistent results.

Figure 10-11 When special cutters are needed, either call a craftsman or become one.

11. Learn to make special cutters. (see Figs. 10-11 and 10-12)

A time will come when "off the shelf" cutters won't do the job. To be truly independent, you should learn to make special cutters. This is where craftsmanship and cutter grinding skills come into play.

I've used the following method as a basis for making many special cutters.

Drill a thru hole the size of the carbide rod near the end of a piece of bar stock, which will be used as the shank of the tool. Mill the end of the bar stock back until about 75% of the hole remains. Stake the pre-split carbide in place with a center punch, silver solder the assembly, then grind about 5º relief and clearance into the carbide. These sturdy cutters can be used as boring bars, undercutting tools, and single flute key cutters.

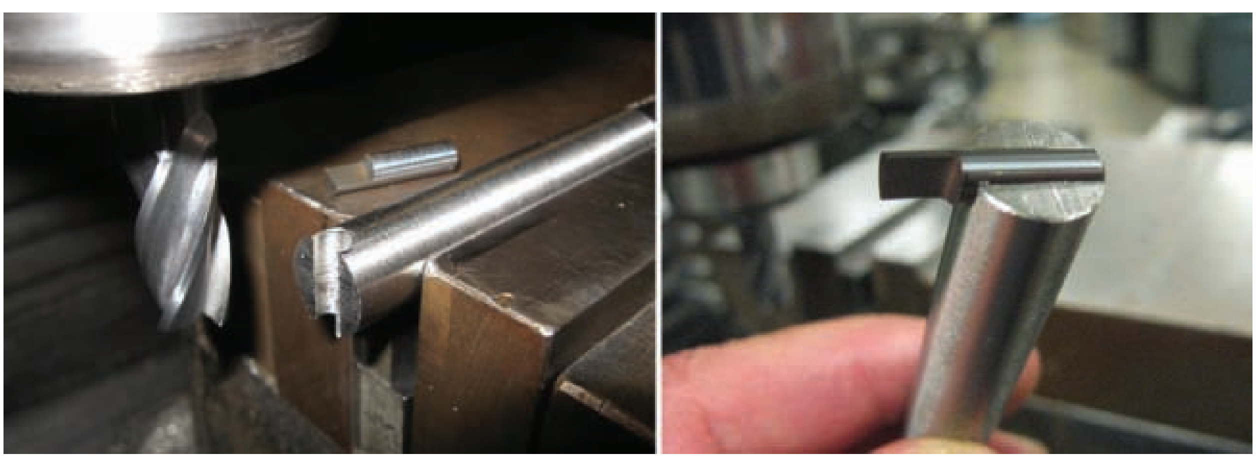


Figure 10-12 Sometimes custom cutters are needed to machine unusual features. These photos show a custom cutter under construction.

12. When making mirrored parts, set the G54 starting point to the center of the part. (see Fig. 10-13)

"Grippers" such as those shown in Figure 10-13 are generally made in pairs — one left and one right. Other parts such as rails, mold bases, and brackets are also commonly used in mirrored pairs. In Figure 10-13, the bottom gripper was the only part programmed. The G54 starting point was set to the middle of the part, which facilitates making a mirror image. With the machine's controller, you simply select which axis you want to mirror along. The figure shows what the mirrored part would look like if the Y axis were selected.

Once selected, run the initial program and a mirror image part will be made. The program will run as if all positive Y values in the program have become negative Y values and vice-versa. You could also select the X axis to mirror along, which would ultimately produce the same mirror image part. The difference is that the program would run as if all positive X values in the program had become negative X values, and so on.

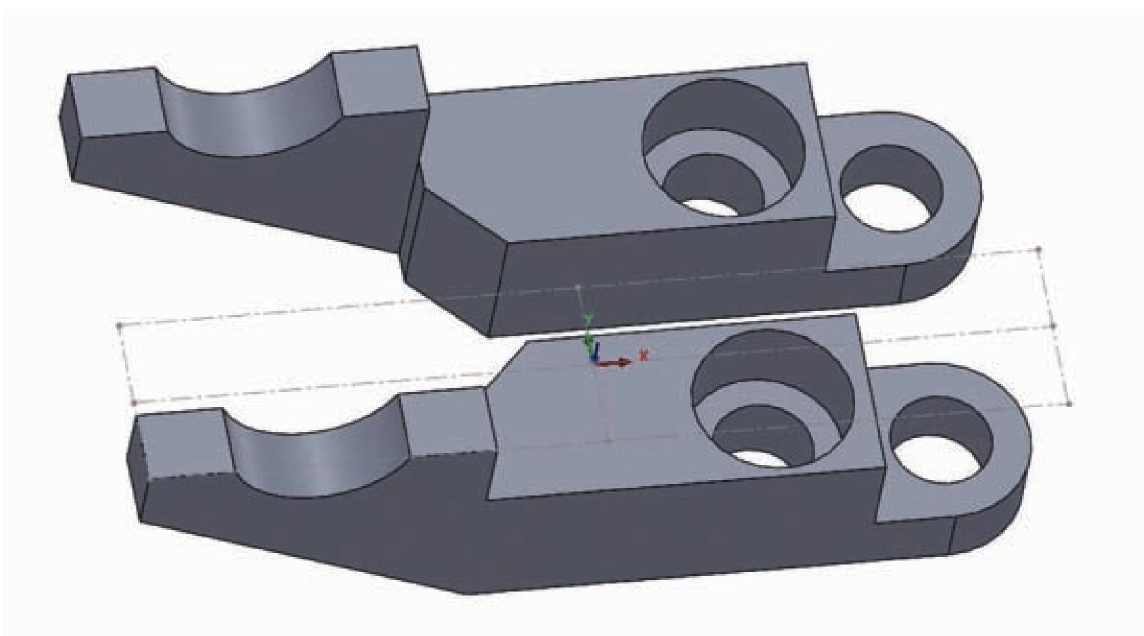


Figure 10-13 Mirrored parts such as these grippers are commonly needed. If you set the G54 starting point to the center of the part, it is a simple matter of activating the mirror feature in the controller to produce a mirror image part.

13. Use 3D milling in a pinch to machine larger countersunk holes. (see Fig. 10-14)

3D milling countersunk holes is not common; it is not an ideal method if you have many holes to countersink. The process is too slow. However, if you don't have a large enough countersinking tool or have just a few holes to countersink, the method can be useful. Large countersinking tools cut with a lot of pressure, which can cause problems such as material bending, tearing, and chatter. For the part in Figure 10-13, there were only four holes so I chose to 3D mill the relatively large counter sinks.

14. When machining flimsy parts, use relatively small diameter end mills and use neutral rake end mills. (see Fig. 10-15)

When I saw these flimsy Delrin parts being run, I was hoping I wouldn't have to get involved. The programmer chose to use large end mills to machine the parts. Initially the job was not going well. Parts were being pulled out of the vise jaws and chunks of Delrin were flying. When I was asked to get involved, I reprogrammed the part using smaller diameter end mills, which cut with less pressure. I also chose to use single flute, zero rake cutters when necessary to avoid the lifting forces inherent with positive rake end mills. My book *Machine Shop Trade Secrets*, 2nd edition, goes into more detail about cutting tools.

Figure 10-14 Though not a common practice, countersinks can be 3D milled. A 1/2" diameter hog nose end mill was used to machine this feature .

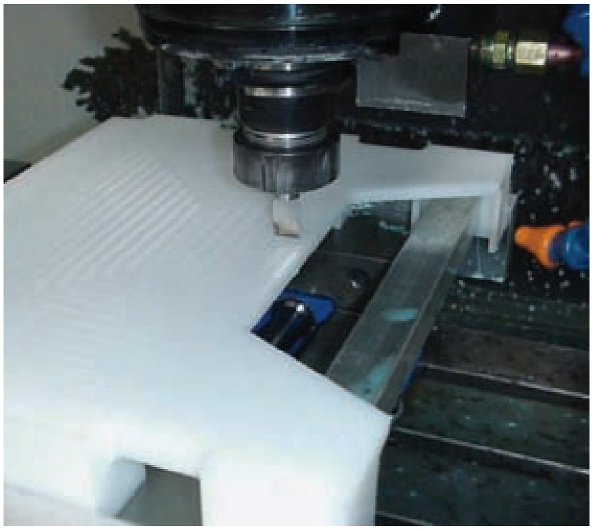


Figure 10-15 Flimsy parts are generally a pain to machine. A relatively small, single flute end mill was needed to finish the upper surface of this part. The part was not rigid enough to use a larger positive rake end mill for this cut.

Figure 10-16 One way to make sure holes are in the correct locations is to set center drills to lightly spot the surface.

15. When verifying a program, set center drills to spot only. (see Fig. 10-16)

Let's face it, we're only human and sometimes we screw up. When drilling a part, especially one that already has a lot of time invested in it, care must to be taken that holes (and other features) end up in correct locations. One way to verify the setup when drilling is to set your center drill to just lightly spot the surface. If the locations happen to be off for some reason, you'll have a chance to correct the error.

16. Set countersink depths manually at the machine. (see Fig. 10-17)

Countersinking tools run the gamut in terms of how they are made. They vary in angle, tool tip size, number of flutes, and more. These variations make it difficult for a programmer sitting at a computer to set a precise depth. The programmer likely won't know what countersink tool will be used because machinists generally grab whatever tool is readily available. Therefore, I don't try at the computer to set a precise depth for a countersink tool. Instead, I program the tip of the tool to stay a little above the part, and let the machinist manually find the correct Z negative value, which can then be entered into the canned cycle that runs the tool.

The easiest way I've found to do this is to run the program until the countersink tool is called up. Then stop the program by hitting the reset button.

After that, start the spindle with the CW (clockwise) button and slowly handle jog the tool to the proper depth in the part. Take note of the depth using the operator's digital readout; enter that value in the canned cycle that runs the tool. The only catch is that the operator's screen must be zeroed in Z beforehand to the tip of the countersink tool or the value shown in the screen won't make sense. Dimensions for countersunk holes are generally not too critical. They are used mostly for flat head screws and lead-ins for tapped holes.

Figure 10-17 Without precise dimensions for a countersink tool, programming a proper depth for the tool would be a guessing game. I've found it easier to let the machinist, at the machine, find the right depth. Once found, the machinist can enter the value in the program.

17. Be especially careful when running large parts with the machine's doors or windows open. (See Fig. 10-18)

Occasionally you'll run into capacity issues with machines. Often these issues can be overcome with thoughtful planning. Sometimes you have to leave the machine's doors or windows open to accommodate a part or some feature of a part. Make a mental note to be extra cautious when doing this. It's a recipe for crashing because your travel may now be limited to the door or window opening and not the machine table's total travel. I saw a nasty crash one time when a machinist had a long plate clamped to the machine table with one end of the plate hanging out the doors. For whatever reason, he homed the machine — and, in rapid mode, the table went home, but not before leaving a big gash in the door as a result of the edge of the plate slamming into it.

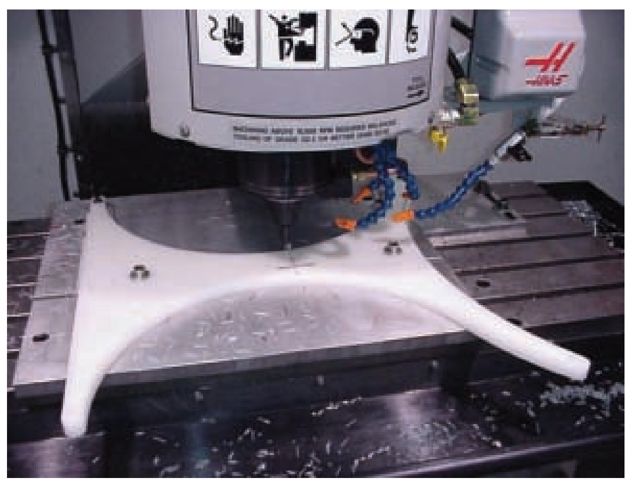


Figure 10-18 Sometimes it is necessary to leave a machine's doors open to accommodate a part. Be extra cautious when doing this because it's easy to forget the travel available may be limited by the opening.

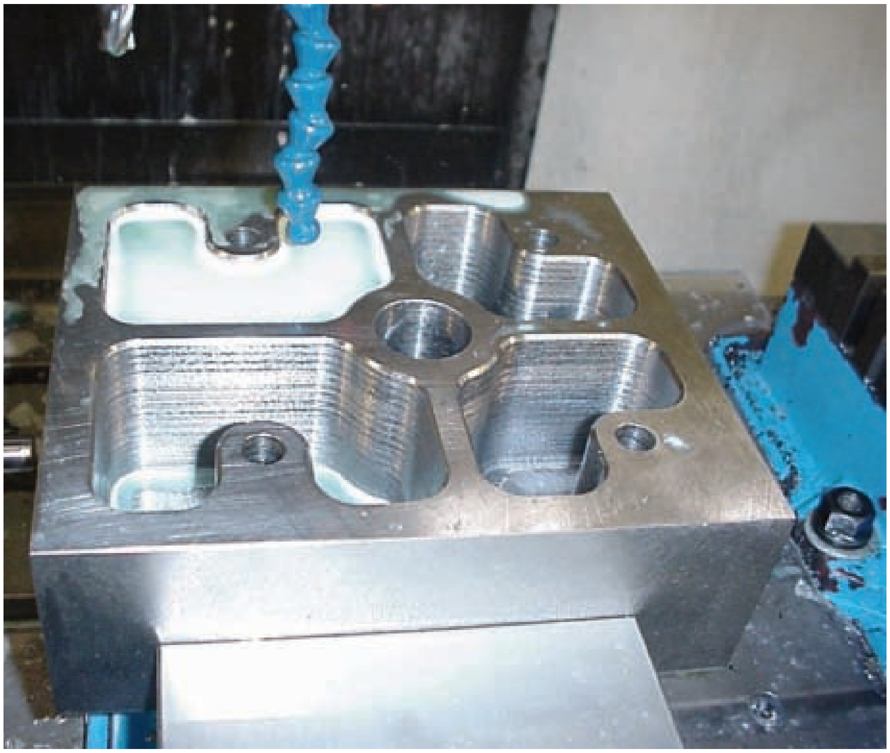


Figure 10-19 The machine's air nozzle is set up to blow trapped chips and coolant from the pockets in this part.

18. Use the machine's air nozzle to blow chips and coolant out of pockets. (See Fig. 10-19)

If your machine is set up with air, use it to your advantage to blow out pockets of chips and coolant. Close the doors and turn on the air. Then handle jog the part under the air stream as needed. You can also add some programming to an existing program to do this automatically. On our machines, air is activated by an M code.

19. Use a grinding vise to hold narrow parts to machine both ends in one setup. (See Fig. 10-20)

The more machining you can do in one setup, the better. Often, short parts can be clamped in a grinding vise mounted in the machine vise to expose both ends for machining. This is a technique I use often.

Figure 10-20 Grinding vises seem to come in handy for a lot of setups. One is being used here to hold a part so both ends of the part can be machined in one setup.

20. When cutting undercuts with keyway cutters, cut the initial slot about .001" deeper than the keyway path. (See Fig. 10-21)

Machining with a keyway cutter can be a little tricky. The cutter is captured in the part and, therefore, has little room to flex. Chips don't have much room to escape. Often you can't see the cut; both the top and bottom of a feature get cut with one cutter. The shank is smaller in diameter than the cutter and is, therefore, inherently weak. You have to make sure the shank has side clearance, you have to allow for enough cutter run off so the cutter won't snag during a tool change, and it's anybody's guess how fast you can feed them. Other than that, they're a breeze to run! Having said that, it's best to start with relatively light chip loads (.001"per tooth) and work up from there.

One technique to avoid rubbing the bottom of a keyway cutter is to cut the center slot (which is cut with a regular end mill) about .001"deeper than the keyway cutter depth.

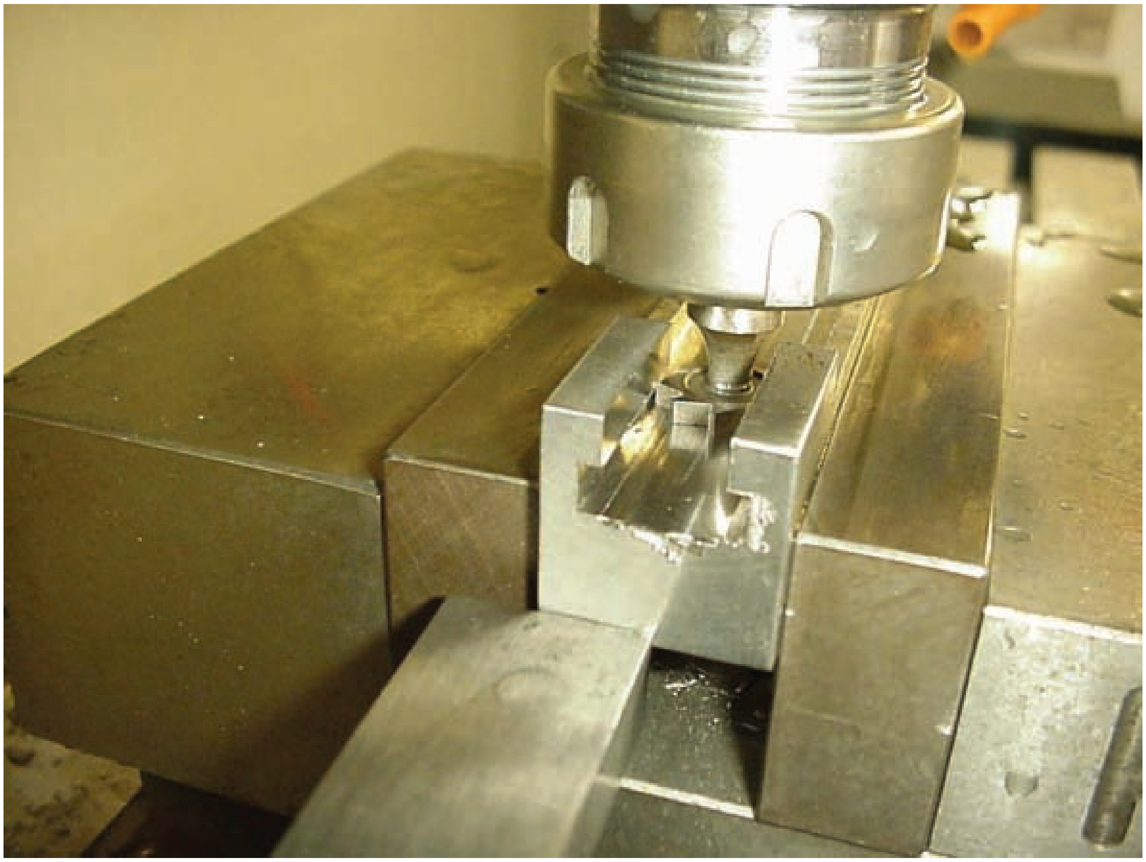


Figure 10-21 The center slot for this part was cut about .001"deeper than the keyway cutter depth. This small bit of clearance helps the bottom of the cutter avoid rubbing and galling.

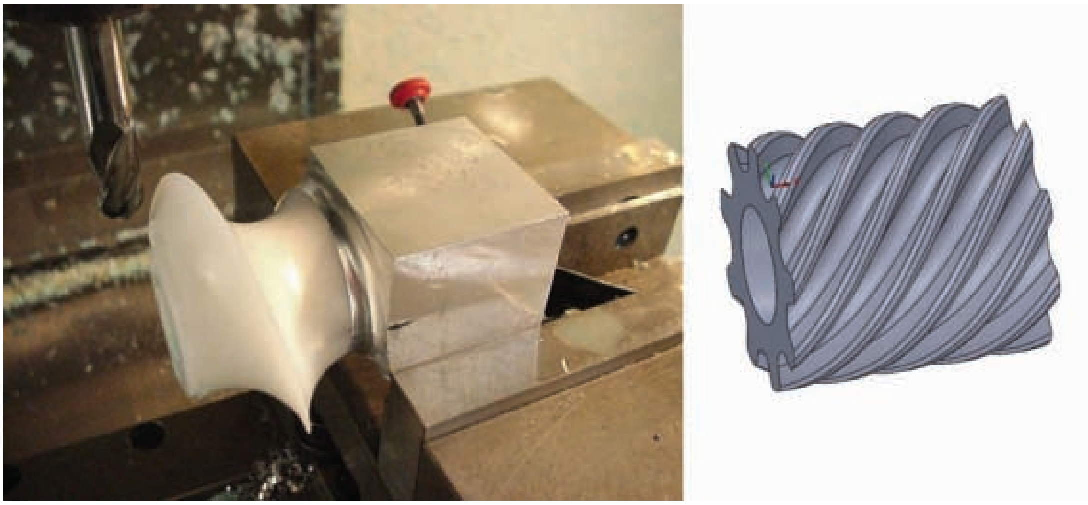


Figure 10-22 Impellor and helical gear-type parts can be made with a 3-axis machine. Four separate programs were used to machine each of these circular parts with each program output to cut the appropriate "top" view. Square holding fixtures were constructed to index the parts around their center axis.

21. Cut circular impellers and gears with a 3-axis CNC machine. (See Fig. 10-22)

When I first saw the drawing for the impellor shown in Figure 10-22, I was skeptical that we could make them. As it turns out, the impellor was machined with little difficulty on a 3–axis machine. The part was made with four setups and four programs. The impellor was first modeled in CAD. Then the CAM software was used to apply 3D cutter paths to four views of the part, each view being ninety degrees apart on the center axis. Blanks were mounted on center to a square holding fixture, as shown in the figure. Armed with four programs, one for each view, I ran the first program, then indexed the block ninety degrees and ran the second program, and so on. The helical gear shown on the right of Figure 10-22 was made the same way.

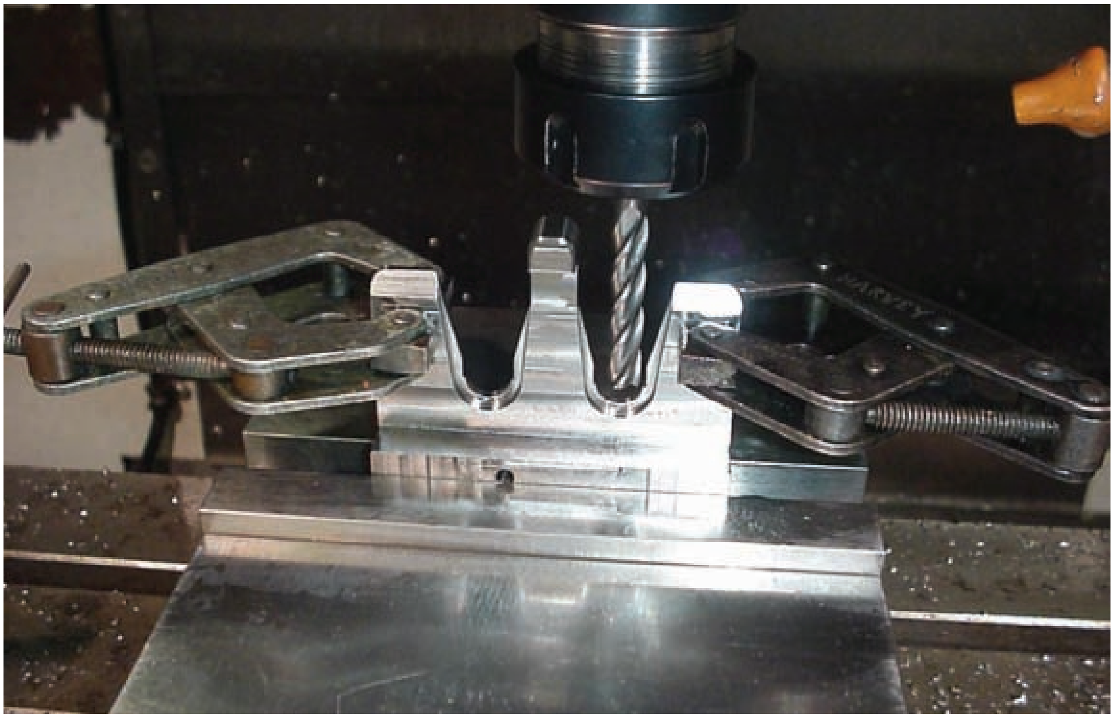


Figure 10-23 Vibration can be reduced or eliminated by adding weight to a part or some feature of a part. In this case, clamps were used to add weight to the outside "ears" of the part, which eliminated the initial vibration.

22. Reduce part vibration by adding clamps. (See Fig. 10-23)

Occasionally the configuration of a part and the type of cut being made will induce unwanted vibration as the part is being machined. The first thing to do is reduce spindle speed or try a different feed rate. The right combination of spindle speed and feed rate often corrects unwanted vibration. In Figure 10-23 neither worked, so I placed clamps used as weights on the two outside "ears" of the part. The change in mass worked wonders for correcting the vibration issue.

That's it. You've come to the end of this introductory book on how to machine parts using CAD/CAM equipment in conjunction with a three-axis milling machine. Machined parts and assemblies are vital to productive manufacturing. When critical parts break, machines stop. That's when machinists are called upon to become saviors. I hope you enjoyed the insights this book provides — they attempt to show in a fundamental way how to become better, more efficient machinists using CAD/CAM equipment.

Suggestions for Odds and Ends

1. Use Silicon mold release on tool holders for easier tool changes.
2. Use Scotch Brite® to remove rust and gunk from tool holders.
3. 3D printing prototype parts can save time.
4. Reduce burrs by re-running the initial facing tool over the part after all features have been machined.
5. Cut hard material with carbide, use no coolant, and have patience.
6. Cutting tools only have to be harder than the material they are machining.
7. Use negative rake end mills to push flimsy parts down onto your parallels and holding fixtures.
8. Use a CNC machine in a pinch to broach blind keyway slots.
9. Experiment with unorthodox cutting routines once in a while.
10. When engraving, make sure the surface being engraved is flat.
11. Learn to make special cutters.
12. When making mirrored parts, set the G54 starting point to the center of the part.
13. Use 3D milling in a pinch to machine larger countersunk holes.
14. When machining flimsy parts, use relatively small diameter end mills and use neutral rake end mills.
15. When verifying a program, set center drills to spot only.
16. Set countersink depths manually at the machine.
17. Be especially careful when running large parts with the machine's doors or windows open.
18. Use the machine's air nozzle to blow chips and coolant out of pockets.
19. Use a grinding vise to hold narrow parts to machine both ends in one setup.
20. When cutting undercuts with keyway cutters, cut the initial slot about .001" deeper than the keyway path.
21. Cut circular impellers and gears with a 3-axis CNC machine.
22. Reduce part vibration by adding clamps.

Index