Thank you for your interest in our CNC curriculums. There is quite a bit of information here, and we appreciate the time you'll take to evaluate it. As you'll see, our curriculums truly minimize the amount of work you'll need to do to get ready to teach a CNC course.

Our method of instruction is proven. It's been developed during over ten years of actual CNC courses - and about half the schools listed in our schools forum (http://www.cncci.com/schools.htm) are using it.

Again, there's quite a bit of information here, so if you have questions, feel free to contact us at any time (email: lynch@cncci.com - or phone 847-639-8847).

Included in this sample package:

Router Programming and Operation

- **Brochure**
  Sales and pricing information about the curriculum
- **Getting started**
  How to use the curriculum
- **Course outline**
  Complete outline of the entire course
- **Sample lesson plans**
  For the first five lessons
- **Sample of student manual**
  For the first key concept - lessons one through seven
- **Sample of the workbook**
  For the first five lessons

From

CNC Curriculums For CNC Instructors!

Machining Center Programming And Operation ♦ Turning Center Programming And Operation

Teaching CNC has never been so easy!

Our CNC curriculums give you a proven, easy to use, concise, yet comprehensive ready-made training program to minimize the preparation you must do to teach CNC courses. Right out of the box you get text books, work books, lesson plans, instructor notes, course outlines, answers to practice exercises, and even software based slide show presentations (in Microsoft Powerpoint vs 4.0) — everything you need to effectively teach CNC courses for the two most popular forms of CNC machine tools in existence today, CNC machining centers and CNC turn-ing centers.

They're FREE with textbook Order!

Not only will you be teaching with the best state-of-the-art CNC curriculums in the industry, you'll be doing so free of charge! All we ask is that your school bookstore buys textbooks/workbooks from us! With an initial order of just 40 text books, we'll ship both complete CNC curriculums free! (20 for one curriculum.) Any combination of machining center or turning center textbooks totaling 40 will qualify. Text books are $50.00 each – work books are $14.95 each. Future orders can be in any quantity. This cost will be recovered, of course, as students enroll in your classes and buy text books. In essence, your first 100 students will be paying for these curriculums!

Who will benefit?

School teachers working for educational institutions will find our course curriculums especially easy to learn and implement. Most CNC teachers come from one of two backgrounds. Either they have extensive CNC experience, but limited teaching experience – or they have extensive teaching experience, but limited CNC experience. These course curriculums will help in both cases.

Our highly structured lesson plans, textbooks, and slide show presentations will make it easy for even an inexperienced teacher to stay on track. And the com-prehensive instructor notes will make experienced teachers who may be a little weak with their CNC skills look like CNC experts!

Instructors working for manufacturing companies will also find these curric-ulums very easy to implement. Few companies have the resources or the desire to develop this kind of program com-pletely from scratch. Additionally, success for this kind of program is not simply a matter of the student getting a good grade. Failure results in scrapped parts, crashed machines, and possibly injured operators. The comprehensive student workbook will prove the student's knowledge of presented materials each step of the way. When the student successfully completes one of these courses, you can rest assured that they comprehend the subject matter.

Training consultants that provide custom training for manufacturing companies will find that this course curriculum makes it easy to teach CNC to their clients. Utilizing standard (and highly portable) computer equipment, the comprehensive slide show can be done on-the-road, meaning instruction can take place anywhere there is a television.

Proven *Key Concepts* approach

Most training experts agree that students learn best when they have a light at the end of the tunnel. The more complex the subject matter, the more important it is that students understand what they must master to successfully complete course. With our curriculums, there are ten key concepts to CNC. The first six are programming-related and the last four are operation-related. Early in the course you can truthfully say "If you can un-der-stand just ten basic ideas, you are well on your way to becoming proficient with CNC."

Another benefit of our key concepts approach is that it lets you explain topics at a broad level — which helps your students get the big picture. With so many different CNC machine and control manufacturers, it's impossible to relate every detail of how each is handled. For this reason, students must have the ability to adapt what they learn to their own machines once they complete your course. In each key concept, you first stress the broad and general reasoning behind how CNC functions, showing students why they must do things as importantly as how to do them. The same reasoning can be applied to any form of CNC machine tool.

Once the student understands this reasoning, you show them specific techniques needed to apply the key concept to the most popular CNC control in the industry. All of the specific examples given in the textbook and slide show are for the Fanuc control. We chose Fanuc as the example control since it is so popular, and since several control manufacturers (Mitusbishi Meldas, Mazatrol, Yasnac, HAAS, Fadal, and others) claim to be Fanuc-compatible.

Here are the ten Key Concepts.

Programming

1 Know your machine (programming)
2 Prepare to write programs
3 Understand the motion types
4 Know the compensation types
5 Know how to format CNC programs
6 Special programming features

Operation

7 Know your machine (operation)
8 Know the three modes of operation
9 Know the key operation procedures
10 Know how to safely verify programs

Lesson structure

The ten key concepts are divided further into lessons. (24 lessons in the machining center curriculum – 28 in the turning center curriculum.) Lessons vary in length based upon complexity and content (from 10 minutes to about an hour).

Machining center curriculum:

1 Machine configurations
2 General flow of programming
3 Visualizing program execution
4 Understanding program zero
5 Measuring program zero
6 The 2 ways to assign program zero
7 Introduction to programming words
8 Preparation for programming
9 Motion types
10 Introduction to compensation
11 Tool length compensation
12 Cutter radius compensation
13 Fixture offsets
14 Introduction to program formatting
15 The four kinds of program format
16 Canned cycles
17 Subprogramming techniques
18 Other special features
19 Rotary tables
20 The control panel
21 The machine panel
22 The three modes of operation
23 The key sequences of operation
24 Program verification

Turning center curriculum:

1 Machine configuration
2 Speeds and feeds
3 Visualizing program execution
4 Understanding program zero
5 Measuring program zero
6 Assigning program zero
7 Flow of program processing
8 Introduction to programming words
9 Preparation for programming
10 Types of motion
11 Introduction to compensation
12 Dimensional (wear) tool offsets
13 Geometry offsets
14 Tool nose radius compensation
15 Program formatting
16 The four kinds of program format
17 Simple canned cycles
18 Rough turning and boring cycle
19 More multiple repetitive cycles
20 Threading multiple repetitive cycle
21 Subprogramming techniques
22 Control model differences
23 Other special features of programming
24 Control panel functions
25 Machine panel functions
26 Three modes of operation
27 The key operation procedures
28 Verifying new programs safely

Student course materials

Student course materials are copyrighted and must be purchased directly from CNC concepts, Inc. A minimum initial order of 100 sets will qualify you to get these curriculums free of charge.

Student textbooks

Our highly tutorial textbooks precisely follows the slide show presentations you will be making. Each is very detailed (over 250 pages) and will provide the student with an excellent way to review the information you present — during the course and long after the course is finished. Of course, as the instructor, you will also want to have one of these manuals handy as you present the course.

Student workbooks

These exercises will confirm that students understand your presentations throughout the course. There is one exercise per lesson.

Instructor course materials

While we do not completely eliminate the preparation you must do to get ready to teach, we really minimize it!

Diskettes of Powerpoint slide presentations

These diskettes of highly compressed files expand to about 16 mega bytes of highly colorful, illustrative, and attention getting Microsoft Powerpoint slide shows (over 1,200 slides each!). To display these slide shows, we even include the Powerpoint Viewer (free distribution copy). However, in order to modify these slide shows, you must have Microsoft Powerpoint software (version 4.0 or higher).

You have several alternatives for projecting the slide shows for your class to see. All involve devices that are connected to the VGA display port of your computer. First, many schools already have a projection system that can display information from a personal computer. Any thing that can be shown on the computer screen (your slide show) can be displayed through the projection system. Second, you can use a device that rests on top of an overhead projector to display your screen shows. Third, and especially if price is a concern, you can use a simple scan converter (about \$200.00 - \$300.00) to display your screen show on any television that has a video-in (RCA) connector.

Instructor Notes manuals

Especially helpful the first few times you teach these courses, these manuals – one for each curriculum – offer many suggestions about how topics should be presented. They make an excellent way to prepare to deliver your presentations. Simply call up the presentation in Powerpoint (or use the Slide Hard Copy Man-ual) to skim through the slides. Having your Instructor's Notes Manual handy while you do so will help you quickly interpret what must be said during each lesson.

Slide show hard copy manuals

These manuals include all slides for each curriculum, helping you to prepare to deliver each lesson and helping you find the slide number for key slides.

Course outlines

These outlines give you a concise way to skim what you will be presenting during the course. Additionally, they make an excellent quick reference for finding the slide numbers in each Powerpoint presentation. At the end of each outline we also show the approximate presentation time you should allow for each lesson.

Blank operation handbooks

In the student textbook for each curriculum is a sample operation handbook for three Fanuc control models. These hand-books will provide students with a good understandings of important procedures that must be documented. However, you will need to document these key procedures for the specific machines in your own facility.

Answers to practice exercises manuals

These answers are provided to help you grade the practices exercises you assign from the student workbook.

Final tests

These tests (one for each curriculum) will help you determine each student's overall understanding of subject matter at the completion of the course.

Getting Started With The Curriculums:

These course curriculums have been designed to address the needs of instructors working in three similar, but subtly different, learning environments. First, instructors working for colleges, vocational schools, technical schools, and universities will find our ready-made course curriculums especially easy to learn and implement. It has been our experience that instructors teaching their first CNC courses for technical schools come from one of but two backgrounds. Either they have extensive CNC experience (possibly through working for a manufacturing company), but limited teaching experience --- or they have extensive teaching experience (teaching other courses for the school), but limited CNC experience. It is rare to find an instructor who has extensive experience in both fields. These course curriculums will help in both cases.

Our proven curriculums, structured lesson plans, and concise lesson format will minimize the amount of work an instructor must do in preparation for teaching a new course -- and minimize the work that must be done when getting ready to teach each lesson *during* the course. The proven *key concepts* approach makes it easy for even an inexperienced instructor to stay on track. And - the slide presentations & comprehensive student manuals will make experienced instructors who may be a little weak with their CNC skills look like CNC experts!

Second, instructors working for companies having their own in-plant training will also find this curriculum easy to implement. Few companies have the resources (or the desire) to completely develop this kind of program from scratch. Additionally, success for an industrial training program is *not* a simple a matter of the student getting a good grade. When finished, students *must* know how to safely program and/or operate the machine tools. Training failure will result in scrapped parts, crashed machines, and possibly even injured operators. The comprehensive student workbooks will prove the student's knowledge of presented materials each step of the way. When the student successfully completes this course, you can rest assured that they comprehend the subject matter.

Third, training consultants that provide custom training for manufacturing companies will find that this course curriculum makes it easy to teach CNC to their clients. Utilizing standard (and highly portable) computer equipment, the comprehensive slide show can be done on-the-road, meaning instruction can take place anywhere.

Five factors that contribute to learning

While experienced instructors may find this information somewhat basic, we wish to ensure that beginning instructors understand the importance of setting up a good learning environment. Of course, the better the learning environment, the better the training program will be, and the faster, and easier it will be for students to learn. This course curriculum is well suited to facilitating the learning environment in all five ways.

Motivation is the most important factor in any learning environment. First and foremost, students must be highly motivated to learn. Motivation will be the driving force that makes students *stick with it* even when they are having trouble understanding information being presented during training. Indeed, *any* problem with learning can be overcome if the student's motivation is high enough. But just as important, the instructor must also be highly motivated to teach. They must have a burning desire to relate information during training in a way students can understand. When student are having problems, the instructor must be motivated enough to spend the extra time it takes to ensure that the student eventually understands presented material. This can be very challenging since students' aptitude levels vary. This course curriculum inspires motivation on both counts. The colorful and illustrative slide show, the tutorial nature of the manual, the pertinent practice exercises, and the highly structured key concepts approach will capture and hold your students' attention, and make it easy for them to *stay* motivated. For the same reasons, instructors should find this course almost *fun to teach* - and it is easy to stay motivated with tasks you enjoy.

Aptitude will determine how quickly and easily learning will be. The aptitude of the instructor for making attention holding presentations, giving pertinent analogies, preparing illustrative visuals, designing realistic practice exercises, and in general, keeping the students interested level high will contribute to determining how quickly and easily students catch on to new material. *Instructors with high aptitude make it easy for students to learn.* In similar fashion, the students' aptitude for learning manufacturing related functions and specifically CNC also contribute to how quickly and easily learning takes place. *Students with high aptitude make it easy for the instructor to teach.* This course curriculum pays particular attention to trouble spots. From our own teaching experience, we know where students with minimal aptitude tend to have the most problems and make ample analogies to keep even the most complex topics of CNC as simple as possible to understand.

Presentation is the heart of training. The better the instructor prepares and delivers the presentation, the easier it will be for students to learn. Presentation can consist of many things, including the instructor's orations, demonstrations, simulations, overhead and projector slides, videos, and anything else that helps to convey an idea. This is the strong point of our curriculums. You will find it very easy to get your ideas across with but a small amount of preparation before delivering each lesson. While you still have to talk, the slide show and instructor notes will make sure you know what to say!

Repetition reinforces a student's understanding of learned information. Even students with extremely high aptitude will find it difficult to learn from presentations made only once. *All* training sessions should begin with a review of recent presentations. Depending upon the frequency and duration of each training session, entire sessions should, at times, be devoted to reviewing what students should already know.
Reviews also help the instructor limit how much complex subject matter is presented during each session. Knowing that certain information will be reviewed, the instructor can avoid getting too deep into complicated topics during the first time the information is presented. Only after

students have a firm grasp of the basics will the instructor dive deeper and introduce more complicated variations. The slide shows really help with review. As you begin each session, you can easily call up the presentation/s made recently and quickly skim material to ensure comprehension. For review purposes, you can even hide slides during reviews (within Powerpoint) to keep from having to go through all but the most important slides.

Practice with reinforcement acts as the gauge to judge the success of training. Well designed practice exercises should be realistic, forcing the student to do things in the same way they must when training is completed. Reinforcement must come as the result of the students' practice. If the student demonstrates a firm understanding of the presented information, reinforcement should praise the success. On the other hand, if practice exposes a student's lack of understanding, reinforcement should come in the form of repeated presentations, review, and more practice, ensuring that the student eventually catches on. This course curriculum include a comprehensive set of practice exercises, as well as a final test, to confirm understanding each step of the way.

The key concepts approach

Most training experts agree that students learn best when they have *a light at the end of the tunnel*. The more complex the subject matter, the more important it is to tell the student early in the training program just what they must master to complete the course. One important benefit of our proven key concepts approach you can do just that. With our curriculum, there are ten key concepts to CNC. Six of the key concepts are programming-related and the last four are operation-related. Early on in the course you can truthfully say *"If you can understand just ten basic ideas, you are well on your way to becoming proficient with CNC."* This gives your students a light at the end of the tunnel and makes learning CNC seem as easy as possible. Also, as you go through the course, students will know where they stand each step along the way.

Another benefit of our key concepts approach is that it lets you work at a very broad level. There are many different CNC machine and control manufacturers making this kind of equipment. Though there are many commonalties in how basic machine utilization is done among current CNC machines, no two machines will be handled exactly the same. For this reason, students will likely have to adapt to their own machines once they complete your course (especially if you teach in a technical school). In each key concept you will be stressing the broad and general reasoning behind how CNC functions, showing students *why* they must do things as importantly as *how* they do them. This reasoning can be applied to any form of CNC machine tool. Once the student understands this reasoning, *then* you will show specific techniques needed to apply the key concept to one very popular CNC control.

By the way, all of the specific examples we give in the manual and during the slide show are in the format for a Fanuc control. We chose Fanuc as our sample control since it is the most popular control in the industry, and since several control manufacturers (Mitusbishi Meldas, Mazatrol, Yasnac, HAAS, Fadal, and others) claim to be Fanuc-compatible. However, you must be prepared for the possibility that your students will be working with a control made by a manufacturer that is not Fanuc-compatible. Again, rest assured that the key concepts apply to every current CNC control on the market. Only the specific techniques needed to apply each key concept must change.

These same ten key concepts can be applied to any form of CNC machine tool. Our course curriculums have been developed around three popular forms of CNC machine tools – for metal cutting, the CNC machining center and CNC turning center – for woodworking, the CNC router (three separate curriculums). While this is the case, the same key concepts can be applied to wire EDM machines, CNC turret punch presses, CNC lasers, and any other type of CNC.

Yet another benefit (from an instructor's standpoint) of the key concepts approach is that the course can end at any time once students understand the key concepts. One major problem in any training program is the limited time available for training. While manufacturing companies may have the luxury of extending a course if students are slow in picking up the material, technical schools commonly work on a very rigid time frame. If students are slow in picking up the material and getting bogged down with the details of CNC machine utilization (asking many time consuming questions), the course may end before you get through all the material. While this presents a problem even with the key concepts approach, the effects of the problem are greatly minimized. As long as students understand the reasoning behind each key concept, it will be just a matter of time until they figure out the details. Given our extensive and highly tutorial course manual, any student with motivation will eventually figure it out (possibly *after* the course ends).

One last benefit we'll mention for the key concepts approach is that it makes reviewing the material easy. Since there are only ten key concepts, they are easy to remember. You can simply restate each key concept and ask pertinent questions to confirm your students retention of material previously presented.

In the lesson plans, student manuals, and course outline, you will find detailed explanations of what must be presented at each key point in the course. Here we simply list the ten key concepts.

Key concept:	Description:
1	Know your machine (from a programmer's viewpoint)
2	Prepare to write programs
3	Understand the motion types
4	Know the compensation types
5	Format your programs in a safe, convenient, and efficient manner
6	Know the special features of programming
7	Know your machine (from an operator's viewpoint)
8	Understand the three modes of operation

9 Know the procedures related to operation
10 You must be able to verify programs safely

Lesson structure

These ten key concepts are divided further concise lessons. For the machining center curriculum, there are twenty-four lessons. With the turning center curriculum, there are twenty-eight lessons. With the CNC router curriculum, there are twenty-three lessons. Lessons do vary in length (from about 10 minutes of presentation to about an hour), and you can find an approximate time of presentation in the course outline.

Machining center curriculum lesson structure

Lesson: Description:

1 Machine configurations
2 General flow of programming
3 Visualizing program execution
4 Understanding program zero
5 Measuring program zero
6 The two ways to assign program zero
7 Introduction to programming words
8 Preparation for programming
9 Motion types
10 Introduction to compensation
11 Tool length compensation
12 Cutter radius compensation
13 Fixture offsets
14 Introduction to program formatting
15 The four kinds of program format
16 Canned cycles
17 Subprogramming techniques
18 Other special features of programming
19 Rotary tables
20 The control panel
21 The machine panel
22 The three modes of operation
23 The key sequences of operation
24 Program verification

Turning center lesson structure

Lesson: Description:

1 Machine configuration
2 Speeds and feeds
3 Visualizing program execution
4 Understanding program zero
5 Measuring program zero

6 Assigning program zero
7 Flow of program processing
8 Introduction to programming words
9 Preparation for programming
10 Types of motion
11 Introduction to compensation
12 Dimensional (wear) tool offsets
13 Geometry offsets
14 Tool nose radius compensation
15 Program formatting
16 The four kinds of program format
17 Simple canned cycles
18 Rough turning and boring multiple repetitive cycle
19 More multiple repetitive cycles
20 Threading multiple repetitive cycle
21 Subprogramming techniques
22 Control model differences
23 Other special features of programming
24 Control model differences
25 Machine panel functions
26 Three modes of operation
27 The key operation procedures
28 Verifying new programs safely

CNC router curriculum lesson structure

Lesson: Description:

1 Machine configurations
2 General flow of programming
3 Visualizing program execution
4 Understanding program zero
5 Measuring program zero
6 The two ways to assign program zero
7 Introduction to programming words
8 Preparation for programming
9 Motion types
10 Introduction to compensation
11 Tool length compensation
12 Router radius compensation
13 Fixture offsets
14 Introduction to program formatting
15 The four kinds of program format
16 Canned cycles
17 Subprogramming techniques

18 Other special features of programming
19 The control panel
20 The machine panel
21 The three modes of operation
22 The key sequences of operation
23 Program verification

Student's Course Materials:
All student course materials are copyrighted and must be purchased from CNC concepts, Inc. Pricing is given on the order form that accompanies this document.

Student Manuals - These highly tutorial manuals precisely follows the slide show presentations you will be making. They are very detailed and will provide the student with an excellent way to review information you present, during the course and long after the course is finished. There is one student manual for each curriculum. Of course, you will also want to have these manuals to use as your master as you present the course.

Student workbooks - This is the set of practice exercises students will be doing during the course. There is one exercise for each lesson (24 exercises for the machining center curriculum, 28 exercises for the turning center curriculum, and 23 exercises for the CNC router curriculum). The practice exercises are like quizzes – you can use them as such or assign them as homework. About half of the practice exercises additionally require the student to do a programming activity.

Instructor's Course Materials:
PowerPoint Viewer diskette - We strongly recommend that you purchase Microsoft PowerPoint (version 4.0 or higher). This will give you total access to all slide show presentations, making it possible for you to add, modify, or delete anything during the slide shows, and custom tailor these curriculums to your liking. However, if you do not have Microsoft PowerPoint, you can use the PowerPoint Viewer to display slide shows. Note that both the PowerPoint Viewer and PowerPoint itself require Microsoft Windows (version 3.1 or higher). To load PowerPoint Viewer, place this disk in the drive A, select RUN from the FILE menu of Windows, type A:\setup, and press enter. Follow the on-screen instructions to complete the install.

Diskettes of PowerPoint Slide Presentations - These disks of highly compressed files will expand to about 10 megabytes for each curriculum on your hard drive. All presentation files will be placed in two subdirectories on your hard drive. For the machining center course, a subdirectory named \MCPO will be automatically created under your root directory. For the turning center course, a subdirectory named \TCPO will be created. For the CNC router course, a subdirectory named \RPO will be created. To load these files to your C:\ drive, place the appropriate diskette (for machining center or turning center curriculum) in the drive and type GO, then press enter. If you wish the files to be loaded to D:\, type GO_D instead (with underscore D). You can then access all needed files in the MCPO, TCPO, or RPO subdirectory right from within PowerPoint (you need not copy them to a PowerPoint subdirectory unless you wish to).

Each of the PowerPoint files includes one key concept. Here are the file names and the related lesson numbers for each curriculum (more comprehensive information is shown in the instructor's notes and instructor outline):

Machining center curriculum files

File:	Lessons:
MCCON1.PPT	1-7
MCCON2.PPT	8
MCCON3.PPT	8
MCCON4.PPT	10-13
MCCON5.PPT	14-15
MCCON6.PPT	16-19
MCCON7.PPT	20-21
MCCON8.PPT	22
MCCON9.PPT	23
MCCON10.PPT	24

Additionally, the machining center curriculum contains a series of animation movie files (in Animation Works format) as well as the movie player to show these movies. While these movies are embedded right in the Powerpoint presentations, they will not be displayed during your presentation unless you have the Animation Works software (most computers do not). Instead, you will have to get the animation player activated before you can display the movies. This can be done as you prepare to deliver any lesson that contains a movie. First of all, here are the movies and their corresponding lessons.

File:	Movie:
DIRMTN.AWM	Directions of motion for a vertical machining center (lesson 1)
HDIRMTN.AWM	Directions of motion for a horizontal machining center (lesson 1)
RAPID.AWM	Example of rapid motion (lesson 9)
FEEDMOVE.AWM	Example program of straight line cutting (lesson 9)
CIRCEXPL.AWM	Example program of circular motion using R word (lesson 9)
IANDJ.AWM	Example program of circular motion using I & J (lesson 9)
TOOLLEN.AWM	Example program of tool length compensation (lesson 11)
CUTCOMP.AWM	Example program of cutter radius compensation (lesson 12)
CANNEDCY.AWM	Example program of canned cycle usage (lesson 16

To load the movies (and the player), place the appropriate disk in drive A and type GO (to load to C:\MOVIES) or GO_D (to load to D:\MOVIES). Once the files are actually loaded, you can the movie player by invoking the file awpa.exe located in the /MOVIES subdirectory. You can do this from the Windows file manager or from the Run selection from the file mode in Windows.

Turning center curriculum files

File:	Lessons:
TCCON1.PPT	1-8
TCCON2.PPT	9
TCCON3.PPT	10
TCCON4.PPT	11-14
TCCON5.PPT	15-16
TCCON6.PPT	17-23
TCCON7.PPT	24-25
TCCON8.PPT	26
TCCON9.PPT	27
TCCON10.PPT	28

CNC router curriculum files

File:	Lessons:
RCON1.PPT	1-7
RCON2.PPT	8
RCON3.PPT	8
RCON4.PPT	10-13
RCON5.PPT	14-15
RCON6.PPT	16-18
RCON7.PPT	19-20
RCON8.PPT	21
RCON9.PPT	22
RCON10.PPT	23

Additionally, the CNC router curriculum contains a series of animation movie files (in Animation Works format) as well as the movie player to show these movies. To display these movies, first activate the animation player. This can be done as you prepare to deliver any lesson that contains a movie. Here are the movies and their corresponding lessons.

File:	Movie:
RAPID.AWM	Example of rapid motion (lesson 9)
FEEDMOVE.AWM	Example program of straight line cutting (lesson 9)
CIRCEXPL.AWM	Example program of circular motion using R word (lesson 9)
IANDJ.AWM	Example program of circular motion using I & J (lesson 9)
TOOLLEN.AWM	Example program of tool length compensation (lesson 11)
CUTCOMP.AWM	Example program of cutter radius compensation (lesson 12)
CANNEDCY.AWM	Example program of canned cycle usage (lesson 16

To load the movies (and the player), place the appropriate disk in drive A and type GO (to load to C:\MOVIES) or GO_D (to load to D:\MOVIES). Once the files are actually loaded, you can the movie player by invoking the file **awpa.exe** located in the /MOVIES subdirectory. You can do this from the Windows file manager or from the Run selection from the file mode in Windows.

Lesson plans - We offer several methods to help you prepare to teach each lesson. Lesson plans (found at the end of this document), will make it easy to understand the key points you must make in order to complete each lesson. In almost every plan, we also offer suggestions about how you can further clarify your points right at your own CNC machine/s. Finally, we offer many suggestions regarding how you can review information and extend the students' understanding once they begin to catch on to basic points.

Instructor's Notes Manuals - While you may find the lesson plans to be sufficient, we also offer many suggestions about how this course can be presented in the instructor notes. It will help you prepare to deliver each lesson and help you locate key slides in the presentation. While these course curriculums dramatically reduce the preparation you must do, we do not completely eliminate it. Until you become *very* comfortable with the course curriculum, we strongly recommend preparing to deliver each lesson. The best way to do so is to call up the presentation in Powerpoint and to skim through the slides. Or if you don't want to fire up the computer, use the Slide Presentation Hard Copy manual to go through the slides. Skimming the Instructor's Notes manual while you do so will help you quickly interpret what must be said during each presentation. As you become more comfortable with the curriculum, you may be able to simply skim the instructor's notes relative to each lesson to get ready to deliver each lesson.

Slide Presentation Hard Copy Manual - These manuals provide you with a quick way to skim each slide in every lesson. Organized by file name (MCCON1.PPT, MCCON2.PPT, MCCON3.PPT, etc.), you'll be able to see exactly what is shown during the your presentations. This should help you as you prepare to deliver each lesson.

Instructor's Outline - Included at the end of this document, this outline gives you a concise way of seeing what you will be presenting during the course. The lessons related to each key concept can be easily determined. Additionally, it makes a quick reference for finding the slide numbers in the PowerPoint presentations. At the end of the instructor outline we also give you approximate times you should allow to make presentations for each lesson.

Sample Operation Handbooks - At the end of each Student Manual is included a sample operation handbook for three Fanuc control models (10M, 11M, and 15M for the machining center curriculum and 10T, 11T, and 15T for the turning center curriculum). This handbook will provide students with a good understandings of important procedures that must be documented.

Blank Operation Handbooks - It is very likely that you do not have one of these controls in your own facility. For this reason, we give you the ability to develop your own operation handbook for the machine/s you do have.

Answers To Practice Exercises Manuals - These answers are provided to help you grade your students performance.

Final Test - This test can be found at the end of this document along with the answers.

What you still need:
In order to show the PowerPoint slide presentations to a group of people, you need the following items.

A computer with Windows 3.1 (or higher) - Just about any current model computer will work. For best results, a 486 or Pentium class is recommended (minimum 4 megs internal). If using a desktop computer, you can easily watch the monitor of the computer (facing your audience) to see the slide show as slides are displayed behind you by the projection system. Since the left mouse button advances the slides, you even have a remote slide advance button (as is commonly used with a 35 mm slide projector). If portability is an issue, keep in mind that many of the notebooks and sub-notebooks have ample power to run the presentation software. However, be careful in your selection. Many notebooks do not allow you to send data out through the VGA port *and* see the slide show on the LCD screen of the notebook at the same time. Without this ability, you may have to turn around to see your slides, which can be distracting to your audience.

Microsoft PowerPoint Software (version 4.0 was used to create the slide show) - Though you can display all presentations with PowerPoint Viewer (included with this curriculum), you will need Microsoft PowerPoint if you intend to modify the slide shows given in this curriculum. We highly recommend that you have this ability. This software can be found in any computer store for a price of about $250.00 (it also comes with *Microsoft Office*). You will find this to be a very powerful presentation generating program; one you can use to develop your own slide shows for other courses (or of course, modify those in this course curriculum).

A way of displaying the screen show - You have several alternatives in this regard. All involve using a device that takes data from the VGA port of your personal computer. First, many schools already have a projection system that can display information from a personal computer. Basically, anything that can be shown on the computer screen can be displayed through the projection system. Second, you can use a device that sits on top of an overhead projector to display your screen shows. In essence, this device makes a transparency of what ever is on the display screen of the computer. Third, and especially if price is a concern, you can use a simple scan converter (about $200.00 - $300.00) and display your screen show on any television that has a *video in* connector (as most do). If you must use the RF connector of the television (where an antenna plugs in), an RF converter must be purchased. Since there are so many alternatives for displaying your slide shows, we welcome phone calls (847) 639-8847 if you have questions about your alternatives.

Putting It All Together

Getting Ready To Teach
As stated earlier, though these course curriculums dramatically reduce the amount of preparation you must do, they do not eliminate it completely. And as any experienced instructor will agree, the key to successful presentations is in becoming comfortable with the material you present. And the only way to get comfortable is through adequate preparation.

Before your first course:
Skim the entire curriculum - Though you do not have to be perfectly comfortable with every detail of the curriculum to begin teaching, you will at least need to understand where the course is going. You can use the course outline, lesson plans, instructor notes, and student manual to gain an appreciation for the ten key concepts and the lesson structure being used.

Before beginning each key concept:
Get comfortable will all lessons in the key concept - While some key concepts have but one lesson, most have more. Be sure you feel comfortable with all points you need to make before you begin teaching. Again, use the course outline, instructor notes, and student manual to increase your comfort level with the entire concept.

Before you deliver a lesson:
Get ready to teach! - Study the lesson plan, instructor notes, and slide presentation hard copy in order to gain an understanding of key points that must be delivered during your presentation.

Practice! - Especially before your first few lessons, get comfortable with your equipment and the material you present by practicing your presentation. In addition to getting you ready to deliver each lesson, this should give you a rough idea of how long it will take to deliver each lesson.

During your presentation of each lesson:
Tell them what you're going to tell them - The lesson plan (key points in the slide show at the beginning of each lesson) will help you prepare your students for what they will be learning. While you don't have to dwell on this slide too long, it will help them know what is coming up.

Tell them - Go though the lesson, using your slide show as a guide. Be sure to point out the page numbers and sections in the student manual where the information is also included for their own independent study. Be sure everyone is catching on. Encourage participation, questions, and comments. While you should find adequate analogies in the slide show to stress the most complex topics, you must be prepared to handle special questions and concerns. Have a blackboard or overhead available for making special points.

Tell them what you told them - The lesson summary (included in the slide show for each lesson) will let you summarize the key points of each lesson.

After you finish each lesson:
Assign and check practice exercises - The students' responses to the practice exercises makes an excellent way to gauge your students' understanding of the subject matter. If you find that students are not doing well, it should be taken as a signal that you must review key information. If students are doing well, be sure to praise them.

As you get deeper into the course:
Review often - No student will retain every word of every presentation you make during a course as lengthy as these. On average, you should spend about 10% to 20% of your session time in review, depending upon how well your students are doing. The more problems they are having, the more time you should spend on review. One excellent way to review is to question students from the previous exercises to confirm their retention.

Let students know where they stand - Be sure everyone knows how they are doing as they progress through the course. Assign special exercises and labs for those students having the most problems. Push those students doing well to go further.

INSTRUCTOR OUTLINE FOR ROUTER PROGRAMMING & OPERATION

Section one - Programming

Slide:	Description (MCCON1.PPT):
1	**Key concept number one - know your machine**
7	Lesson structure
	Lesson 1 - Machine configurations
15	Lesson plan
16	Basic machining practice discussion
27	Vertical machining centers
28	Components
33	Direcitons of motion
35	Horizontal machining centers
36	Components
41	Directions of motion
52	Other programmable features of machining centers
53	Lesson 1 summary
	Lesson 2 - General flow of programming
55	Lesson plan
57	Get the big picture
59	Decision is made as to which CNC machine to use
60	The machining process is developed
61	Tooling is ordered and checked
62	The program is developed
63	Setup documentation is made
64	Program loaded into the CNC control
65	The setup is made
66	The program is cautiously verified
67	Production is run
68	Corrected version of the program is stored for future use
69	Lesson 2 summary
	Lesson 3 - Visualizing the execution of a CNC program
72	Lesson plan
73	Analogy to giving a set of instructions
91	Program make-up
92	Command structure
97	Word structure
104	Sequential method of program execution
105	Visualizing movements
127	Other notes about program structure
128	Decimal point programming
133	Decimal point omission
134	Modal words
136	Initialized words
138	Word order within a command
140	Lesson 3 summary

Slide:	Description (MCCON1.PPT):
	Lesson 4 - Understanding program zero and the rectangular coordinate system
143	Lesson plan
144	What is program zero?
145	Graph analogy
148	Base lines
149	Origin of graph
150	Axes of a machining center
162	XY plane
164	Increments of each axis
165	Plus and minus coordinates
167	XZ plane
169	Example of coordinates relative to program zero
178	Example of coordinates in Z
181	Where to place the program zero point
201	Understanding the incremental mode
212	Advantages of the absolute mode
215	Incremental warning
219	Lesson 4 summary
	Lesson 5 - Measuring the program zero point
222	Lesson plan
223	Why program zero must be measured
228	What is zero return position?
233	How the measurement is made
234	For rectangular workpieces in the X and Y axis
255	For round workpieces in the X and Y axis
266	For the Z axis
276	How to avoid program zero measurements
277	Lesson 5 summary
	Lesson 6 - The two ways to assign program zero
280	Lesson plan
281	Limitations of older controls
283	Assigning program zero in the program
286	Polarity of dimensions
291	The G92 command
294	The major problem with G92
299	A safety command for use with G92
300	Assigning program zero with fixture offsets
302	Polarity of dimensions
307	The fixture offset page of the display screen
310	Commands related to fixture offsets
311	How fixture offsets are programmed
312	Advantages of fixture offsets
313	Safety
314	Multiple workpieces

Slide:	Description (MCCON1.PPT):
315	Rerunning tools
316	Efficiency
320	Lesson 6 summary

Lesson 7 - Introduction to programming words (

323	Lesson plan
324	How many words are used?
327	O word
328	N word
329	G word
330	X word
332	Y word
333	Z word
335	A, B, & C words
336	R word
338	I, J, & K words
339	Q word
340	P word
343	L word
345	F word
346	S word
347	T word
348	M word
349	D word
350	H word
351	End of block (EOB) word
352	Slash (/) code
353	Points about G words
357	Points about M codes
360	Points about other programming words
364	Lesson summary

Key concept number two - Preparation for programming (MCCON2.PPT)

Lesson 8 - The importance or preparation

2	Lesson plan
3	Introduction to key concept number two
4	Preparation and safety
14	Steps for preparing to write a CNC program
22	Prepare the machining process
23	Reasons to do the process up-front
30	Do the required math and mark-up the print
31	Marking up the print
32	Doing the math
33	Check the required tooling
38	Plan the set-up
40	Lesson summary

Slide:	Description (MCCON3.PPT):

Key concept number three - Types of motion commands (MCCON3.PPT)

Lesson 9 The three kinds of motion

2	Lesson plan
3	Introduction to key concept number three
4	What Is Interpolation?
7	Linear interpolation introduction
8	Circular interpolation introduction
9	Helical interpolation introduction
10	The four types of motion
15	Common functions of all motion types
21	Understanding the point being programmed
62	G00 Rapid motion (also called positioning)
37	Examples of rapid
39	G01 linear interpolation (straight line motion)
44	Example program showing straight line motion
46	G02 and G03 Circular motion commands
53	Example program showing circular motion
55	Using directional vectors (I, J, & K)
56	I, J, & K defined
57	Examples of I, J, & K
59	Example program using I, J, & K
61	Arc limitations
71	Making a full circle in one command
74	Example of full circles in one command
78	Example program making full circle
87	Helical motion
87	Example of motion
88	Lesson summary

Key concept number four -The three kinds of compensation (MCCON4.PPT)

Lesson 10 - Introduction to compensation

2	Lesson plan
3	Introduction to key concept number four
10	Reasons for compensations
15	Marksman analogy
20	Understanding offsets
22	The offset pages of the control
24	Instating offsets
29	Lesson summar

Lesson 11 - Tool length compensation

32	Lesson plan
33	Introduction to tool length compensation
36	The reason for tool length compensation
48	Programming words for tool length compensation
54	When to instate
59	Example commands to instate

Slide:	Description (MCCON4.PPT):
61	Do you need to cancel?
63	The two methods to use tool length compensation
69	Using the tool length as the offset
76	How to measure the tool length on the machine
90	How to measure tool lengths off line
94	Summary of key points
100	Example program showing tool length compensation
102	Using the distance from the tool tip to program zero as the offset
108	How to measure the offset
121	Advantages of method number one (tool length as offset)
125	Final notes about tool length compensation
129	Lesson summary

Lesson 12 - Cutter radius compensation

132	Lesson plan
133	Introduction to cutter radius compensation
135	When to use
136	Reasons for using cutter radius compensation
137	Making calculations easier
167	Range of cutter sizes
189	Workpiece sizing
194	Roughing and finishing
195	Words used with cutter radius compensation
200	The two ways to use cutter radius compensation
203	When the cutter radius is stored as the offset value
204	Steps to using cutter radius compensation
208	Initializing
209	G41 or G42?
227	How to determine the offset number
235	How to position the tool prior to instating
239	Example of instating
246	Making movements with cutter radius compensation
258	Cancelling cutter radius compensation
262	Example program showing cutter radius compensation
268	Using the center line path (as CAM systems generate)
284	Lesson summary

Lesson 13 - Fixture offsets

287	Lesson plan
288	Introduction to fixture offsets
291	Benefits of fixture offsets over G92

Slide:	Description (MCCON4.PPT):
297	Words used with fixture offsets
299	Using only one fixture offset (use G54)
301	Using fixture offsets for more than one coordinate system
314	Warning about Z
318	Two ways to input fixture offsets
320	Unknown distances from one program zero point to another
325	Known distances from one program zero point to another
331	Using fixture offsets on horizontal machining centers
342	G53 Machine coordinate system command
347	G10 Data setting by program command
360	Lesson summary

Key concept number five - Program Formatting (MCCON5.PPT)

Lesson 14 - Introduction to program formatting

2	Lesson plan
3	Introduction to key concept number five
5	Four styles of program formatting
11	Safety versus efficiency
17	Machine differences that affect program formatting
23	M codes
25	Automatic tool changer
36	Reasons for formatting programs
60	Lesson summary

Lesson 15 - The four types of program formatting)

63	Lesson plan
64	How to use the formats
80	Format styles
81	For vertical machining centers using G92 to assign program zero
84	G28 explained
127	Rerunning tools when using G92 to assign program zero
132	For vertical machining centers using fixture offsets
176	Rerunning tools when using fixture offsets to assign program zero
181	For Horizontal Machining Centers Using G92 to Set Program Zero
228	For Horizontal Machining Centers Using fixture offsets to Set Program Zero
273	Example program showing format information
276	Lesson summary

Slide:	Description (MCCON6.PPT):

Key concept number six - Special features of programming

Lesson 16 - Introduction to special programming features and canned cycles

2 Lesson plan
3 Introduction to key concept number six
9 Canned cycles for hole machining operations
10 Reasons for canned cycles
12 Things all canned cycles have in common
19 Kinds of canned cycles
20 G73 Peck drilling cycle for steel (chip breaking)
35 G74 Left hand tapping cycle
42 G76 Fine boring cycle with no drag line
51 G80 Canned cycle cancel command
55 G81 Standard drilling cycle
62 G82 Counterboring cycle
70 G83 Deep hole drilling cycle (with pull out)
83 G84 Right hand tapping cycle
90 G85 Reaming cycle (feed out)
98 G86 Standard boring cycle (leaves drag line)
105 G89 Counterboring cycle for boring bar (with dwell)
112 Words used in canned cycles
125 Understanding G98 and G99
135 Three examples showing the use of canned cycles
177 Extended example showing canned cycle usage
179 Notes about tapping
188 Speeds and feeds for tapping
193 What is rigid tapping?
201 Canned cycles and the Z axis
220 Using canned cycles in the incremental mode
261 Lesson summary

Lesson 17 Subprogramming techniques

263 Lesson plan
265 When to use subprogramming techniques
272 How subprograms work
283 Words involved with subprograms
292 Example applications
292 Multiple identical machining operations
329 Multiple machining operations on holes
344 Rough and finish contour milling
360 Multiple identical workpieces
365 Summary of advantages
369 Nesting subprograms
374 Using M99 in a main program

Slide:	Description (MCCON6.PPT):

379 Using subprograms as control programs
380 With pallet changers
385 Lesson summary

Lesson 18 - Other special features of programming

388 Lesson plan
389 Introduction to other special features
391 G04 - Dwell command
396 Applications for dwell
397 To relieve tool pressure
403 To program around machine problems
410 To force sharp corners
416 G09 and G61 - Exact stop check
432 G50 and G51 - Scaling commands
443 G60 - Single direction approach mode
450 G68 and G69 - Coordinate rotation
462 The slash code (/) - Optional block skip command
463 How optional block skip works
467 Trial boring to size a hole
470 Other applications for optional block skip
476 G50.1 and G51.1 - Mirror image commands
497 Thread milling
512 Types of thread milling cutters
524 Programming commands
541 Understanding your G and M codes handbook
543 Lesson summary

Slide: Description (MCCON6.PPT):

Section Two: Operation

Key concept number seven - Know your machine (MCCON7.PPT)

Lesson 19 - The control panel

Slide	Description
2	Lesson plan
4	Introduction to operation
6	Introduction to key concept number seven
7	The operator's responsibilities
11	The directions of motion from an operator's viewpoint
34	The two most basic operation panels
36	The control panel
36	Basic layout
37	Letter address keypad
39	Display screen functions
40	Soft key functions
41	Position function
47	Program function
50	Offset Function
52	Program check function
54	Setting function
55	Message function
56	Hard keys for display screen functions
57	Lesson summary

Lesson 20- The machine panel

Slide	Description
60	Lesson plan
62	Machine panel layout
65	Other buttons and switches you may have
74	Other panels
75	Other operator functions
83	Lesson summary

Key concept number eight - The three modes of machine operation (MCCON8.PPT)

Lesson 21The three modes of operation

Slide	Description
2	Lesson plan
3	Introduction to key concept number eight
6	The manual mode
11	The manual data input mode
15	The program operation mode

Slide: Description (MCCON8.PPT):

Slide	Description
18	Lesson summary

Key concept number nine - The key sequences of machining center operation (MCCON9.PPT)

Lesson 22 - The key sequences

Slide	Description
2	Lesson Plan
3	Introduction to key concept number nine
8	Understanding the operation handbooks
9	Manual sequences
11	To start the machine
16	To do a manual zero return
33	To manually start the spindle
36	To manually jog axes
41	To use the handwheel
44	Sequence to manually load tools into the spindle
47	Sequence to load tools into the tool changer
50	To manually turn on and off the coolant
54	To set the axes displays
61	To enter and change tool offsets
67	To manually activate mirror image
74	To select inch and metric modes
80	Manual data input sequences
82	To use MDI to change tools
89	To use MDI to activate spindle
95	To use MDI to do a zero return
99	Program loading and saving and sequence
101	To load programs by tape
107	To load programs through the keyboard
111	To send programs from the control
116	Program editing and display sequences
134	Setup sequences
135	To measure the program zero point
179	To measure tool lengths
192	Lesson summary

Key concept number ten - Program verification (MCCON10.PPT)

Lesson 23 - Program verification procedures

Slide	Description
2	Lesson plan
3	Introduction to key concept number ten
5	Safety priorities
9	Common mistakes
14	Procedures to program verification
15	The machine lock dry run
19	Free flowing dry run
23	Normal air cutting cycle execution
27	Running the first workpiece
31	Running verified programs

Slide: Description (MCCON10.PPT):

Slide	Description
32	Rerunning tools
33	Lesson summary

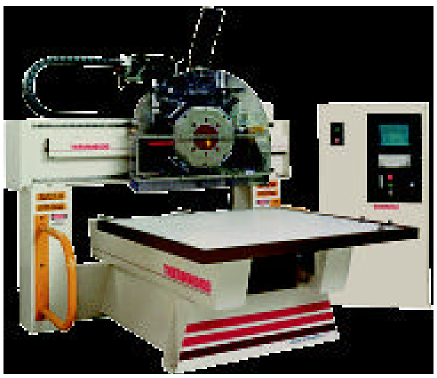
List Of Lessons:

Here is a listo of all lessons and the presentation time it should take you to make the presentation for each lesson (not including questions and comments from your students). Obviously, no to instructors will present information in exactly the same manner, so these are only approximate times. Also, no time is allowed here for the practice exercise (possibly given in the form of homework or quizes).

Lesson	Min.	Description
Lesson 1	30	Machine configurations
Lesson 2	15	General flow of programming
Lesson 3	30	Visualizing program execution
Lesson 4	30	Understanding program zero
Lesson 5	30	Measuring the program zero point
Lesson 6	30	Two ways to assign program zero
Lesson 7	30	Introduction to programming words
Lesson 8	30	Preparation for programming
Lesson 9	60	Three kinds of motion commands
Lesson 10	20	Introduction to compensation
Lesson 11	50	Tool length compensation
Lesson 12	50	Cutter radius compensation
Lesson 13	40	Fixture offsets
Lesson 14	30	Introduction to program formatting
Lesson 15	40	The four kinds of program format
Lesson 16	70	Canned Cycles
Lesson 17	40	Subprogramming techniques
Lesson 18	40	Other special features
Lesson 19	40	Rotary devices
Lesson 20	30	The Fanuc control panel
Lesson 21	30	The Machine panel
Lesson 22	25	The three modes of operation
Lesson 23	40	The key sequences of operation
Lesson 24	40	Program verification techniques

Programming and Operating CNC Routers

Lesson Plans



CONCEPTS, INC.

Machine configurations 1

Approx. presentation time: 20 minutes — Activities: Lecture, lab, exercise

Introduction to key concept number one:

Since this is your very first presentation, you'll want to make your students feel welcome and comfortable. Explain general considerations like course times, break times, and any other preliminary information you wish to relate. Explain the key concepts approach and lesson structure you'll be using. Acquaint students with the course materials (text book and work book), and discuss any other materials you will be using. Point out any materials you expect them to bring to each class (calculator, note-taking materials, etc.). If there are any prerequisites students must understand before taking this course, be sure to mention them. Encourage participation in the form of questions and comments. Make any other points you wish to make about your learning environment. It's also helpful to have students introduce themselves. You can get a good initial gage of their level by having them talk about their current level of CNC experience - and youmay find someone who has extensive experience that can help you make your points throughout the course.

Key concept number one: *Know your machine from a programmer's viewpoint*. It is, of course, important that all students get off on the right foot. Be sure to let them know that your using a building blocks approach. If they do not understand early presentations, they'll have problems later. Have them question anything they do not understand. Use the slide show to introduce the seven lessons related to key concept number one. The main thrust of this key concept is to get beginning programmers understanding the things they must know about CNC routers in order to write CNC programs.

General:

Lesson one is intended to help beginners understand the basic machiing practice related to woodworking as well as the configuration of CNC routers. While there are a variety of CNC router configurations, be sure to point out that they share many things in common. Throughout this course, you'll be stressing the most popular: the *bridge style* CNC router.

Key points to make:

Basic machining practice: Though it has little to do with this course curriculum, be sure students understand the importance of basic machining practice as it relates to the kind of machine they will be programming. The more they know about machining practice, the easier it will be to learn CNC programming. Since most CNC routers are multi-tool machines, and since it is not uncommon for a CNC router program to utilize many tools, CNC programmers must also know how to *process* workpieces to be machined.

Variety of CNC routers available: While this lesson introduces a variety of CNC router configurations, be sure students understand that you will be concentrating most on the bridge style CNC router for much of the course. Be sure to point out similarities among the various types (among other things, axes remain the same from the perspective of the spindle nose, X is always left/right, Y is fore/aft, and Z is spindle in/out). Also explain the three most popular applications for bridge style CNC routers (flat board work).

Major components: While CNC programmers do not have to be machine designers, it doesn't hurt to familiarize your students with the most basic machine components, especially those that are programmable. At the very least, include discussions on headstock & spindle, workholding devices, table, tool changer, vacuum system and CNC control. You may also wish to explain the workings of axis drive systems, including drive motors, ball screws, and way systems.

Axes of motion: Explain each axis. If your machines have a rotary axis, be sure to include it in your discussion. Make sure students understand the directions of each axis, including plus and minus.

Warning about plus versus minus: Though you may not want to get too bogged down in this presentation if students are having trouble understanding this point, mention that there is a discrepancy regarding how programmers should view axis motion plus and minus versus the way operators view plus and minus. Point out that it is easiest for programmers if they thing of plus versus minus as if the *tool* is actually moving in all axes - and it is in all three axes with bridge style routers (tool moving to the right is X+, tool moving away is Y+, and tool moving up is Z+). You must specify, however, that for some CNC routers , it is not the tool that actually moves to form the axis. Instead, the *table* motion forms the X and Y axes. Explain that fro these machines, the table must move to the left in order for the tool to move toward the right, meaning *table movement toward the left is X plus motion*. Warn beginners that this is a common source of confusion between programmers and operators, since CNC operators tend to look for table motion to determine plus/minus as opposed to tool motion. Back off if beginners are having problems understanding this. Explain that you'll cover this in more detail later in this key concept while discussing program zero.

Other programmable functions: Let students know that with full blown CNC routers, almost anything they need to control can be controlled through programmed commands. While you do not want to get bogged down explaining the details of how each device is programmed, at least mention programmable devices like spindle, feedrate, coolant, automatic tool changer, pallet changer, and any other programmable function on your machine/s. I like to briefly introduce the programming word/s related to each device at this point. Also point out that not all machine functions are programmable with some CNC routers. The lower the cost, the fewer the number of features that will be programmable.

How speed and feed are programmed: Point out that most CNC routers only allow one way to control speed and feedrate. Spindle speed is programmed in RPM (in one-RPM increments) and feedrate is specified in per minute fashion (inches or millimeters per minute). This means, of course that programmers must calculate spindle rpm and inches (or millimeters) per minute feedrate. They must, of course, understand basic machining practice related to cutting conditions in order to do so.

Discuss the machine's zero return position: Explain that the zero return position (also called grid zero, machine zero, or home position) is the machine's reference position. Normally located close to the plus overtravel limit of each axis, show students this location for each of your machines.

Other things to do:

- Take your students out to see your CNC router/s. Show the components and demonstrate the directions of motion. Especially if table motion makes up one or more of the machine's axes, demonstrate plus versus minus. If you have a program ready to run, show them how a program is run. Keep in mind that the whole idea is to stress the points made so far. Don't get bogged down with questions on other topics.

Other notes for this lesson:

General flow of the programming process 2

Approx. presentation time: 10 minutes Activities: Lecture, exercise

General:

This short lesson is intended to help your students get the big picture of what goes on in a company's CNC environment. Of course, this course is highly focused on programming and operation of CNC routers. But by giving your students the ability to step back and see the bigger picture, they will better understand their role in the CNC environment.

Key points to make:

Flow of programming process: Use the slide show and text book to illustrate what goes on in the CNC environment from the decision as to which CNC machine will be used for a given operation to storing the corrected version of a verified program once production is running.

Stress how programs can be loaded: While you may elect to have your students typing programs into the control through the keyboard for the sake of getting practice, be sure students understand the various methods companies use to upload and download programs.

Other notes for this lesson:

Visualizing the execution of a CNC program 3

Approx. presentation time: 20 minutes Activities: Lecture, lab, exercise

General:

At this early stage in the course, beginners will still be somewhat intimidated by CNC equipment. If your demonstration of a CNC CNC router in lesson one was a student's first exposure to CNC machine tools, they will likely think that most of what they have seen is almost magic. In this lesson, you will remove the mystery from how CNC machines function, letting students know that CNC programming is highly structured and logical - *not magic*.

Key points to make:

A programmer must be able to visualize movements: This point can not be overstressed. Without this ability, the programmer can not write CNC programs. Stress that this is why basic machining practice is so very important. But even good woodworkers can sometimes have trouble visualizing. Explain that a person with good basic machining practice experience already knows *what* they want the machine to do. It will be a relatively simple matter to learn *how to tell* the machine what it is they want it to do. Visualizing the tool's movements during the cutting of a workpiece is of paramount importance when it comes to telling the machine what it is you want it to do.

Give an analogy to travel instructions: In the slide show we emphasize this need for visualization by giving a simple analogy. Before a person can write down a set of travel instructions, they must first be able to visualize the path from the start point to the end point. Only then can a set of step-by-step instructions be written. In similar fashion, a CNC programmer must be able to visualize the step-by-step manner that a workpiece is machined before a CNC program can be written.

Compare a CNC program's execution to running a manual machine: We give an example during the slide show and in the book for drilling a hole on a drill press. Use this example to stress how to visualize a program's motions. Any experienced machinist could easily list the step by step procedures (especially if standing right in front of the machine) needed to drill a hole on a drill press (turn spindle on, position tool, turn on coolant, plunge hole, retract, etc.). However, when sitting behind a desk, armed with little more than print, paper, pencil, and calculator, visualizing these steps is not so easy. But this kind of visualization is mandatory in order to write CNC programs.

Explain the make-up of a CNC program: The slide show and text book offer very good examples of a CNC program's structure. Stress how programs are made up of *commands* (or blocks). Compare the commands of a CNC program to the sentences in the English language. Explain that, like sentences, CNC commands are made up of *words*. Point out that each CNC word is made up of a *letter address* and a *numerical value*. Make sure students understand that just as sentences in the set of travel instructions tell a person how to get from point A to point B, CNC commands tell the CNC machine how to machine a workpiece.

Explain the sequential order of CNC program execution: Point out that the CNC control will execute a CNC program in exactly the same manner a person will follow a set of travel instructions - *sequentially*. With the travel instructions, first the person will read and interpret the first instruction and do what it tells them. Only then will they proceed to the next instruction. In similar fashion, the CNC control will read, interpret, and execute the first command in the CNC program. Only then will it move on to the next command. (Actually, just as the person following travel instructions, it will start interpreting the next command as it executes the current command.) Point out that just as a person will follow the set of travel instructions to the letter, so will a CNC control execute a program exactly as it is written. Just as a bad instruction in the set of travel instructions results in a person getting lost, incorrect commands in the CNC program results in incorrect motions.

There is only one result per CNC command: Explain that the English sentences given in the set of travel instructions are subject to misinterpretation. Misinterpretations lead to several potential results to a given instruction. Explain that this is *not* the case with the execution of a CNC program. Regardless of the order of CNC words within a CNC command, the control will only have one resultant interpretation of what will be done. In this sense, CNC programs are rigidly structured.

Other things to do:

- Since this lesson includes a complete CNC program (in the visualization analogy), you may want to pause long enough to discuss some of the actual words involved in programming. Though you must point out that you don't expect students to remember all of what you're saying (yet), it's good to expose them to CNC programming words in the context of a CNC program as early in the course as possible.
- At the very least, explain decimal point programming and the meanings of words like modal and initialized. Show why decimal points are so important within certain CNC words by showing the older fixed format and how the control will interpret X, Y, and Z coordinates if the decimal point is not included in the word.
- To stress the sequential order of CNC program execution, demonstrate the running of a CNC program on your own machine while the machine is under the influence of single block. Get the program up on the display screen and execute it line by line, showing your students how the program is executed step-by-step.

Other notes for this lesson:

Understanding program zero 4

Approx. presentation time: 20 minutes Activities: Lecture, exercise

General:

This lesson is the first of three that deals with program zero. Students must understand how positioning movements are commanded for CNC machine tools. While experienced CNC people take this for granted, beginners may find this concept a little difficult to comprehend. Begin by asking this question: *"How many revolutions of an axis drive motor or ball screw does it take to cause one inch of linear motion?"* Of course no one will be able to answer, but this should get them thinking about how motion is generated. Explain that while very early NC machines (over 30 years ago) required this kind of cryptic programming, thanks to a feature called *program zero* we no longer have to be concerned about the internal workings of the machine to cause precise axis motion. It's handled on a much higher level.

Key points to make:

Give the graph analogy: In the slide show and text book we show an excellent analogy that should help you easily explain program zero. Everyone has had to create or interpret a graph. Use this prior knowledge to explain the basics of what program zero is all about. Discuss that like the base lines of the graph, CNC machines have linear axes. Just as the base lines of a graph represent something (time and productivity in our example), so do the axes of a CNC machine represent an actual point in space. Just as base lines are broken into increments, so are axes broken into linear measurement increments (in inches or millimeters). And most importantly, point out that just as a graph will have an origin (where the base lines cross), so will each axis have a program zero point. Make sure students understand that program zero is the origin for positions to be included within the CNC program. Instead of having to specify the number of revolutions of a drive motor or ball screw, the CNC programmer will specify movements by designating positions relative to program zero. And instead of plotting points that represent conceptual ideas, CNC programmers will be plotting physical locations in space for tool movements.

Discuss the smallest increment for programming: As stated, each axis is broken into measurement increments. Point out that the input mode (inch or metric) determines how small the increments will be. In the inch mode, the smallest increment is 0.0001 in for most CNC routers. Point out that this means a ten inch long linear axis has 100,000 positions!

Discuss plus and minus: Point out that in the graph analogy, it just so happened that all points fell up and to the right of the origin. Be sure your students know that this is but one quadrant in the coordinate system. Points plotted in this area are plus in both axes. However, with CNC programming, it is likely that points must be plotted in other quadrants. Use the drawings in the textbook and slide show to illustrate how to determine the polarity of points plotted in other quadrants.

Warn beginners about forgetting the point of origin for polarity: It is not uncommon for beginners to have problems understanding how to determine a point's polarity. Be sure students understand that the program zero point is the point of origin for all movements, and the sign of the coordinate is determined accordingly. To elaborate, demonstrate times when a positive movement must be made to get to a negative position (the starting point of the motion is yet more negative than the ending point).

Where to place the program zero point: Be sure beginners understand that the placement of program zero is very flexible, and commonly this placement is at the discretion of the programmer. Use the slide show and manual to demonstrate that the dimensioning techniques on the workpiece drawing determine where program zero is placed. The origin of each axis (datum surface) on the print makes an excellent program zero point. Explain that the wise placement of program zero results in many of the coordinates going into the program coming right from the drawing. Also point out that some design engineers do not dimension in a logical manner. If having trouble determining the best program zero location, have beginners ask themselves from which surfaces the workpiece will be located in the setup. Location surfaces make an excellent program zero point.

Explain absolute versus incremental programming: Stress that whenever using program zero as the point of origin for causing movements, the programmer is working in the absolute mode. Be sure beginners understand that this is the best mode for programming and that a G90 command is used on most CNC routers to designate absolute positioning movements. However, beginners should also understand that movements can be made incrementally (by designating G91). Explain that the tool's current position is the point of origin when working in the incremental mode. Use the slide show and manual to demonstrate the differences.

Discuss the advantages of the absolute mode: Use the slide show and textbook to describe why beginners should concentrate on exclusively using the absolute mode (coordinates are easier to calculate, coordinates make sense, mistakes are not compounded, etc.). While you may want to mention applications for incremental programming (most having to do with subprogramming techniques), urge beginners to work exclusively in the absolute mode until they are very comfortable with programming.

Warn beginners about thinking incrementally: Warn your students that beginners have the tendency to think incrementally. When programming movements, they tend to ask themselves how far should the tool move. Get them to ask *to what position* should the tool move. In the absolute move, this position is always relative to program zero.

Other things to do:

- Have students practice determining the best location for program zero based on a variety of prints.
- Have students practice determining the coordinates for points needed within CNC programs with a series of practice prints.

As they catch on:

- When it comes to the smallest increment for programming, you stressed that in the inch mode, the smallest increment is 0.0001 inch. While most companies in the US still work in the inch mode, you may want to make students aware that in the metric mode, the smallest increment is 0.001 millimeters. 0.001 mm is actually less than half of 0.0001 inch, meaning the machine has a finer resolution in the metric mode. In fact, the same ten inch long linear axis that has 100,000 positions in the inch mode has 254,000 positions in the metric mode! Explain that this gives the CNC user the ability to hold tighter tolerances in the metric mode, since positions can be programmed to a finer resolution and offsets can be modified in smaller increments.

Other notes for this lesson:

Measuring the program zero location 5

Approx. presentation time: 20 minutes Activities: Lecture, lab, exercise

General:

Students must understand that just because they wish the program zero point to be in a given location, the control will not automatically know the position of this location. By one means or another, program zero must be *assigned*. Explain that this assignment marries the CNC program to the setup in the CNC router, and in essence, calibrates the machine with the tools held in the spindle. While there are many ways to actually assign program zero, and while some are much more efficient that those we show, our given methods are based on ease-of-understanding, and beginners should readily catch on to what the program zero assignment represents. Do point out that the methods shown can be improved upon as they catchon.

Key points to make:

Understanding the zero return position: Use the slide show to illustrate that the machine's zero return position is the reference point of the program zero assignment values. That is, the measurements are taken between program zero to the spindle while the machine is resting at the zero return position.

The program zero setting values: While there are some exceptions, use the slide show and textbook to explain that many CNC routers require that the *distance between program zero and the spindle (center in X/Y, nose in Z)* be determined in order to assign program zero. One value is required for each axis, and we'll call these values the program zero setting values.

One way to determine the program zero setting values is to measure them: Use the slide show and textbook to show one method of measuring the program zero assigning values. Be sure students understand that this is a rather crude way of doing it, but it makes it easy to visualize what the program zero assigning numbers represent. In reality, there are special techniques and machine functions that can be used to streamline - if not eliminate - these measurements. Point out that during these measurements, the machine is simply being used as a measuring device.

What about variations in tool length: As stated, the program zero assigning values are specified to the center of the spindle in X/Y. However, explain that in the Z axis there are actually two popular ways to determine the program zero assignment value, based on how tool length compensation is used. Though tool length compensation is discussed until key concept number four, explain how the Z axis program zero assigning value is determined now based upon how you intend to teach tool length compensation. If you use our recommend method of teaching tool length compensation, each tool's length will be stored in a tool length compensation offset. In this case, the Z axis program zero assigning value will be the distance between program zero and the spindle nose while the machine is resting at its zero return position. If you teach the secondary method of using tool length compensation (distance from each tool tip down to program zero is offset), the Z axis program zero assignment value will be zero.

The program zero point assignment simply targets the tooling: Be sure students understand that even if measurements are made very cautiously, it is likely that discrepancies in the *static* measurements they make from the actual *dynamics of cutting* a workpiece will cause minor variations in the first workpiece being machined. That is, just because the program zero point is correctly measured is no guaranty that each tool will cut perfectly. Additionally, other problems caused by tool wear will mean more variations in workpiece size during the production run. Point out that these problems are handled by other means (tool length compensation, cutter radius compensation, and trial machining), and that while program zero assigning numbers should be as accurate as possible, they are really only targeting the position of program zero.

Other things to do:

- While this is handled in much greater detail during the operation section of the course, it wouldn't hurt to go out to the machine and demonstrate the actual techniques used to measure the program zero assigning values for a simple setup.

As they catch on:

- During reviews of this material, push what students already know about program zero assigning values a little further. Ask if they can come up with any suggestions for minimizing the number of measurements that must be made to determine program zero assigning values. Examples include:

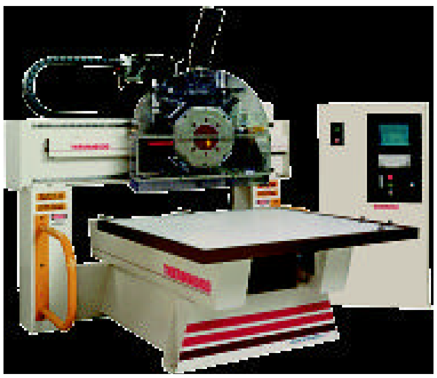
  No need to remeasure the program zero point the next time the setup is made if the setup is qualified (locating fixture/vise from the machine's table slots).

  By knowing some constant values, it will be possible to *calculate* the program zero assigning values, eliminating measurements altogether for qualified setups.

Other notes for this lesson:

Programming and Operating CNC Routers

and woodworking machining centers




Published By:

CNC CONCEPTS, INC.

Written by Mike Lynch

Document number: S00041

Machining Center Programming Quick Reference Card

G Codes

Code	Description	Status	Initial-ized	Modal
G00	Rapid motion	Std	Yes	Yes
G01	Straight line cutting motion	Std	No	Yes
G02	CW circular cutting motion	Std	No	Yes
G03	CCW circular cutting motion	Std	No	Yes
G04	Dwell	Std	No	No
G09	Exact stop check, one shot	Std	No	No
G10	Offset input by program	Opt	No	No
G17	XY plane selection	Std	Yes	Yes
G18	XZ plane selection	Std	No	Yes
G19	YZ plane selection	Std	No	Yes
G20	Inch mode	Std	Yes	Yes
G21	Metric mode	Std	No	Yes
G22	Stored stroke limit instating	Opt	No	Yes
G23	Stored stroke limit cancel	Opt	Yes	Yes
G27	Zero return check	Std	No	No
G28	Zero return command	Std	No	No
G29	Return from zero return	Std	No	No
G30	Second reference point return	Opt	No	No
G31	Skip cutting for probe	Opt	No	No
G40	Cancel cutter radius comp.	Std	Yes	Yes
G41	Cutter radius comp. left	No	No	Yes
G42	Cutter radius comp. right	No	No	Yes
G43	Instate tool length comp. (+)	No	No	Yes
G44	Instate tool length comp. (-)	No	No	Yes
G45	Tool offset expansion	No	Yes	Yes
G46	Tool offset reduction	Std	No	Yes
G47	Tool offset double expansion	Std	No	Yes
G48	Tool offset double reduction	Std	No	Yes
G49	Cancel tool length comp.	Std	Yes	Yes
G50	Cancel scaling	Opt	Yes	Yes
G51	Scaling on	Opt	No	Yes
G52	Return to base coord. system	Opt	Yes	Yes
G53	Shift to mach. coord. system	Std	No	No
G54	Instate fixture offset #1	Std	No	Yes
G55	Instate fixture offset #2	Std	No	Yes

Code	Description	Status	Initial-ized	Modal
G56	Instate fixture offset #3	Std	No	Yes
G57	Instate fixture offset #4	Std	No	Yes
G58	Instate fixture offset #5	Std	No	Yes
G59	Instate fixture offset #6	Std	No	Yes
G60	Single direction positioning	Opt	No	Yes
G61	Exact stop check mode	Std	No	Yes
G64	Normal cutting (cancel G60&G64)	Opt	No	Yes
G65	Custom macro call	Opt	No	No
G66	Custom macro modal call	Opt	No	Yes
G67	Cancel custom macro call	Opt	Yes	No
G68	Coordinate system rotation	Opt	No	Yes
G69	Cancel rotation	Opt	Yes	Yes
G73	Chip breaking peck drilling	Std	No	Yes
G74	Left hand tapping cycle	Std	No	Yes
G76	Fine boring with no drag line	Std	No	Yes
G80	Cancel canned cycle	Std	Yes	Yes
G81	Drilling cycle	Std	No	Yes
G82	Counterboring cycle	Std	No	Yes
G83	Deep hole peck drilling cycle	Std	No	Yes
G84	Right hand tapping cycle	Std	No	Yes
G85	Reaming cycle	Std	No	Yes
G86	Boring cycle	Std	No	Yes
G87	Back boring cycle	Std	No	Yes
G88	Boring cycle	Std	No	Yes
G89	Boring cycle with dwell	Std	No	Yes
G90	Absolute mode	Std	No	Yes
G91	Incremental mode	Std	Yes	Yes
G92	Program zero designator	Std	No	Yes
G98	Return to initial plane (G73-G89)	Std	Yes	Yes
G99	Return to rapid plane (G73-G89)	Std	No	Yes

Notes about G codes: 1) Machine tool builders vary dramatically with regard to which G codes they make standard. 2) Parameters control the initialized state of certain G code groups (like G90-G91). 3) Not all control models include all G codes shown in this list.

Common M codes

Code	Description	Status	Initial-ized	Modal
M00	Program stop	Std	No	No
M01	Optional stop	Std	No	No
M02	End of program (no rewind)	Std	No	No
M03	Spindle on forward (CW)	Std	No	Yes
M04	Spindle on reverse (CCW)	Std	No	Yes
M05	Spindle off	Std	No	Yes
M06	Tool change command	Std	No	No
M07	Mist coolant	Opt	No	Yes
M08	Flood coolant	Std	No	Yes
M09	Coolant off	Std	Yes	Yes
M19	Spindle orient	Std	No	Yes
M30	End of program (rewinds)	Std	No	No
M98	Subprogram call	Std	No	No
M99	Subprogram return	Std	No	No

Other M codes you may have

Code	Description	Status	Initial-ized	Modal
____	Pallet change	___	___	___
____	Chip conveyor on	___	___	___
____	Chip conveyor off	___	___	___
____	Hydraulic clamp on	___	___	___
____	Hydraulic clamp off	___	___	___
____	Indexer rotation	___	___	___
____	____________	___	___	___
____	____________	___	___	___
____	____________	___	___	___
____	____________	___	___	___
____	____________	___	___	___
____	____________	___	___	___
____	____________	___	___	___
____	____________	___	___	___

As with G codes, M code numbers vary dramatically from one machine tool builder to another. Be sure to check the M codes list that comes with your machine to see what other M codes you may have.

Table of Contents

Part One - Programming

Part Two - Setup and Operation

Course Description and Orientation

It is the intention of this course to introduce beginners and experienced programmers alike to programming, setup, and operation techniques used with CNC routers and woodworking machining centers. We will begin in a basic manner, ensuring that even newcomers to CNC will be able to follow all presentations. As the course progresses, we will constantly build upon previously presented information, making it easy (even for the complete novice to CNC) to follow along with the course with a minimum of effort. This building blocks approach will be used for the entire course, and when finished, you will have a thorough understanding of what programming, setup, and operation are all about.

Key concepts approach

Our *key concepts approach* will keep a light at the end of the tunnel, letting you know where you stand as you progress. They will also help you keep your thoughts organized, giving you a proven way to learn about CNC.

There are ten key concepts you must master to become proficient. Key concepts one through six are related to programming, while key concepts seven through ten are related to setup and operation. For programming, key concepts numbered one through four form the most important building blocks, giving you a way to organize your thoughts for this sophisticated type of equipment. Key concepts five and six put it all together.

Beginners should concentrate most on understanding the key points made early on during each key concept. *It is just as important to know why you are doing things as it is to know how to do them*. Concentrate first on the whys. It will be impossible a novice to totally memorize and comprehend every technique used in programming the first time it is presented or read. Rest assured that if you can understand the basic reasoning behind why each CNC feature is used, it will be *much* easier to learn how to use the feature. Once this basic reasoning is understood, it will be easy to go back and review the information to extract the pertinent details of how each CNC feature is used so you can start putting your CNC router or woodworking machining center to good use.

For experience programmers, this course will provide you with an excellent way to compare our recommended programming methods to what you may be accustomed to. If you have had previous CNC programming, setup, or operation experience, you will find it relatively easy to adapt what you already know to the presentations we make.

As experienced programmers know, there are many ways to format CNC programs. This course will show you one or two safe ways to accomplish your programming tasks *that work*. You can use your own common sense and past programming experience to develop your own programming style.

Programming style

We'll be teaching *manual programming*, which is also referred to as *G code level programming*. Admittedly, it is the most rudimentary form of CNC programming, and many companies have other ways to develop CNC programs. Many companies, for example, have computer aided manufacturing (CAM) systems. These systems allow programmers to draw the workpiece shape (or import it from a computer aided design [CAD] system). Then the cutting operations can be commanded. When finished, the CAM system will output a G code level program (much like the programs you'll be learning how to write in this course).

Other companies have *conversational CNC controls* that allow shop floor programming. You can think of a conversational control as like having a single-purpose CAM system built right into the CNC control. Like CAM systems, most will create a G code level CNC program that is used to drive the machine.

Regardless of how CNC programs are prepared, it is quite important that everyone involved with CNC equipment (programmer, setup person, operator) understands how commands are given at G code level for three reasons. First, for simple work, manual programming may be the *best* alternative for program development. In some cases, developing the CAM system program is just as difficult and takes just as long. It also requires that the programmer learn the CAM system as well as the CNC machine.

Second, CNC programs must often be modified right at the machine. At the very least, it's likely that cutting conditions will be changed from time to time. Worse, mistakes will have to be corrected. The person making these changes must, of course, understand G code level programming to make the necessary changes.

And third, it is during a discussion of manual programming that we can best introduce CNC-related features. These are features that must also be understood, regardless of how programs are developed.

Prerequisites

Rest assured that this course will cover CNC machining centers from the ground up. We will assume that the student has *absolutely* no previous experience with CNC. However, there are certain things we do assume about students taking this course.

Basic machining practice experience

We will assume that you have at least some experience with basic machining practice as it applies to woodworking. While lesson one introduces many of the cutting tools used in the woodworking field and discusses some of the basic terminology, it is not intended to replace a full course on woodworking. It will be helpful, for example, if you understand the workholding tooling related to woodworking, the related cutting tools, and how to determine cutting speeds and feedrates based upon the material you must machine. And we assume you understand how to develop the tooling sequence (process) for jobs that require multiple tools.

If you have previous experience with any form of manual woodworking equipment (routers, jointers, planers, table saws, drill presses, etc.), we think you will find it remarkably easy to learn how to program and operate a CNC router or woodworking machining center. Think of it this way: You already know *what* you will be wanting the machine to do. It will be a relatively easy task to *learn how to tell the CNC machine* to perform your desired operations. This is why woodworkers make the best CNC programmers, setup people, and operators.

On the other hand, if you have no previous woodworking experience, or worse, no shop experience whatsoever, your task will be much greater. You not only need to learn CNC programming, setup, and operation, you also need to learn the basic machining practices that go into woodworking. At the very least, you will need the help of an experienced woodworker to explain the operations you will be performing on the CNC router. More likely, you'll need training in this area. If you have no previous woodworking experience, we strongly recommend that you enroll in a such a training program in conjunction with this course. These courses can be found at your local technical/vocational schools, colleges, and universities. We cannot stress this enough: The more you know about woodworking, the easier it will be to learn CNC programming, setup, and operation!

Math

The word *numerical* in computer numerical control implies that numbers are highly involved with CNC. Indeed, every CNC command includes numbers, and almost every CNC command requires an arithmetic calculation to be made. However, most calculations are quite simple. The types of arithmetic calculations required for the typical CNC router or woodworking machining center program includes simple addition, subtraction, multiplication, and division. For more complex workpieces, some right-angle trigonometry may also be required.

This course will require very little in the way of math (though if you are using this manual in conjunction with a *live* course, your instructor may wish to include more math). We simply assume that the student can add, subtract, multiply, and divide. We will be teaching CNC router usage, not mathematics. For this reason, our examples will be quite simple with regard to math. However, we do not wish to understate the importance your math capabilities. In real life, many CNC programmers are required to perform some rather complex calculations including trigonometry in order to come up with coordinates needed in CNC programs. Or, some form of computer aided manufacturing (CAM) system must be used to reduce the amount of math required for programming.

Motivation

This should almost go without saying. We assume that your are motivated to learn. If you are highly motivated to learn about CNC woodworking equipment, it will make your task much easier. Your motivation will overcome *any* problems you have with learning. With motivation, you'll stick to it until you understand.

Controls covered

Since the Fanuc control is the most popular CNC control available, specific examples will be given in Fanuc format. Keep in mind, however, that the key concepts approach we use throughout this course will make it possible for you to learn techniques that can be applied to just about any control on the market. Also keep in mind that several control manufacturers claim that their controls are *Fanuc compatible*. These manufacturers include Yasnac, Mitsubishi Meldas, and Tasnac. Even if you do not have one of these controls, we strongly feel that if you understand the basic concepts, and if you understand how specific techniques are applied to one control type, it will be relatively easy to adapt what you know to just about any CNC router control on the market.

When it comes to machine tool operation, the specific techniques used to operate the various control models mentioned above vary dramatically. For this reason, we include a set of *Operation Handbooks* for the various controls this course covers. These operation handbooks will show you the key operation procedures required for CNC operation in *step-by-step* fashion. If you are using this manual in conjunction with a formal CNC course, your instructor may have developed special operation handbooks for the machines owned by your school or company.

Controls other than Fanuc

Though the techniques given during this presentation are specific to Fanuc controlled CNC machining centers, keep in mind that most CNC router controls are programmed with very similar techniques. For this reason, this course will work nicely to present information required even for programming controls other than Fanuc.

Instruction method

Note that this manual is intended to be used with a formal CNC course. Possibly you are enrolled in a technical school's CNC course. Or maybe you are attending a company's in-plant CNC training course. In these cases, you have an instructor making presentations and available to help you understand the material. On the other hand, you may be using this manual in conjunction with a *video course*, meaning your instructor is on tape. Either way, oral presentations should help you easily understand the concepts. And with video, you can rewind and review until you understand.

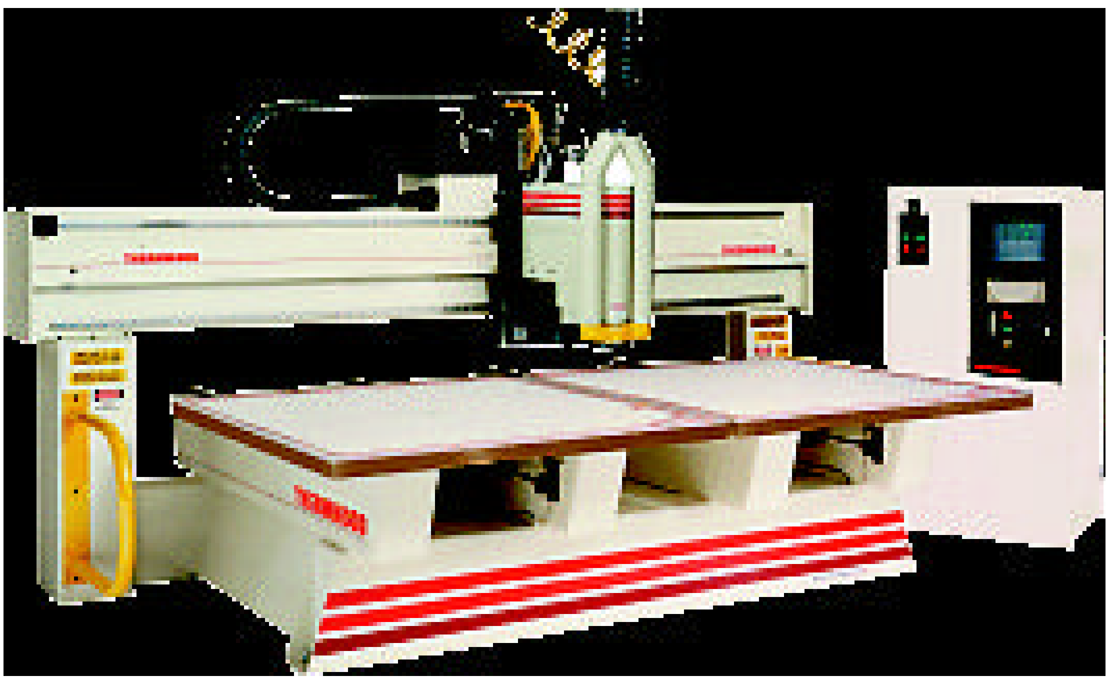
If you have purchased this manual separately and do not have the benefit of an instructor, your task will be a little harder. While most of the key points made during live presentations are included in this text, you will be left on your own to study hard enough to grasp the material presented.

Good Luck!

We at CNC Concepts, Inc. wish you the best of luck with this course. We hope you find it easy to understand our written presentations and the oral presentations of your instructor (live or on video tape). Once completed, we hope this course makes your introduction to Fanuc controlled CNC routers as easy and enjoyable as possible.

Section Number One:

Programming



Key concept number one - Know your machine

The first key concept is *you must understand the basic make-up of the CNC machine tool you will be working with*. Though this may sound like a simplistic statement, there are actually two vantage points for understanding your CNC router or woodworking machining center. To become fully versed with the machine, you must understand the machine from a programmer's viewpoint as well as from an operator's viewpoint. Since we begin with programming, we will now discuss those things about the CNC router with which a programmer must be acquainted. Much later, during the operation portion of this course, we will discuss those things about your CNC router a setup person or operator must know.

Lesson 1

1.1. Basic machining practice related to CNC routers and woodworking machining centers

The single-most important topic a CNC router programmer *must* understand is related to the basic machining practice of woodworking operations. The more a beginning programmer knows about woodworking, the easier it will be to become proficient as router programmer, and the better programmer they will be.

From a beginner's standpoint, a CNC router center can be easily compared to a manual hand held router. A router bit is rotated at a fast speed and driven into contact with the material being routed. But instead of depending upon a person' steady hand for motion control, the CNC router has the ability to precisely drive the router with electrical and mechanical systems. Additionally, some CNC routers have the additional ability to automatically change cutting tools (router bits, drills, and other woodworking cutting tools). This gives them the ability to perform several operations on a workpiece in a completely automatic fashion. Since these versatile CNC routers can perform so many different operations (not just routing, they are called *machining centers*. Many machining centers can completely machine a workpiece, regardless of how many tools are required during the CNC cycle.

Think of it this way. A person who already has a good knowledge of woodworking already knows *what* they want the CNC router to do. It will be a relatively simple matter to learn the various programming commands needed to tell the CNC router *how* to perform the desired operations. This is why woodworkers make the best CNC router programmers. With a good understanding of woodworking under your belt, learning to program CNC routers is actually quite easy.

If you have experience with hand held routers, table saws, drill presses, and other woodworking machinery, and if you understand the related cutting tools, believe it or not, you are well on your way to understanding how to program a CNC router. Your previous experience has prepared you for much of what you need to program a router.

On the other hand, if you have little or no woodworking experience, unfortunately, your task will be much greater. You will not only have to learn the basics of CNC, you will also have to learn what CNC routers are intended to do. Compare this to learning how to fly an airplane without understanding the basics of aerodynamics and flight. Any pilot would agree that a student must understand the theory behind flight before they can get into the cockpit.

While many of the principles of woodworking are relatively simple to understand, if you have limited woodworking experience, we strongly recommend that you enroll in one of the excellent courses offered for woodworking at a local technical school. Or, if there is someone in your company who is well versed and is willing to share their knowledge, by all means, take advantage of this help. At the very least, pick up a beginner's book on woodworking (there are many available). For without a good understanding of the operations a CNC router is intended to perform, the novice will be in for a great deal of frustration.

We also compare the importance of knowing woodworking in order to write CNC programs to a person making a speech having to be well versed in the topic being presented. If the speaker is not well versed, they will not be able to make much sense during their presentation. In the same way, a CNC router programmer who is not well versed in router-related cutting operations will not be able to prepare programs that make any sense to experienced woodworkers.

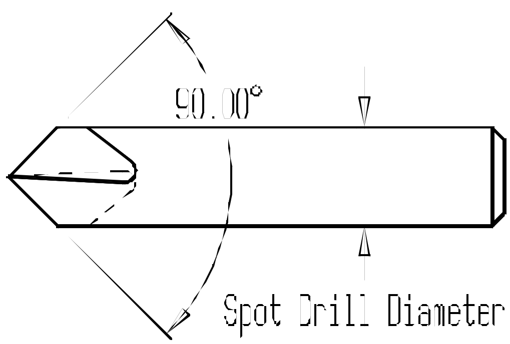
What follows is *not* intended to replace a full course on woodworking. Instead, it is for those students who may be a little weak in their woodworking skills or need a refresher in this area. In this section, we offer a *brief* discussion of the most common cutting operations performed on CNC routers and provide some basic terminology. We additionally offer suggestions for any special programming implications that might be involved for specific cutting operations. Again, *we are not trying to eliminate the need for you to learn more about woodworking*. We are simply trying to acquaint you with the specific cutting tools and machining operations performed on CNC routers that we *assume* you understand as this course is presented.

1.1.1. Hole machining operations

Hole machining is commonly performed on CNC routers, especially those equipped with automatic tool changing devices (machining centers). For this reason, most CNC router control manufacturers give you a series of special *canned cycles* to help with hole machining operations. While canned cycles are discussed much later in the course (during key concept number six), we mention them here to let you know that there are very simple commands available to help with the programming of hole machining operations. In this discussion, we intend to acquaint you with the many basic machining practice points related to hole machining operations as well as what you need to know in order to develop your CNC program.

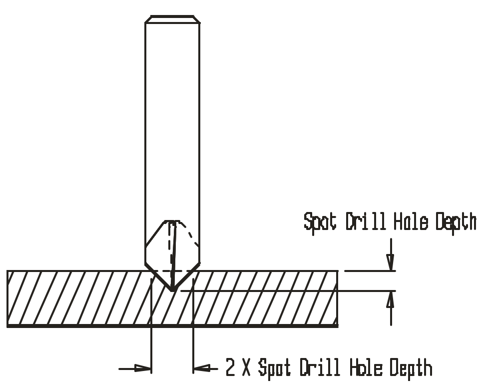
1.1.1.1. Spot drilling

One way of premachining a hole prior to drilling is spot drilling. It is often necessary to make a chamfer around the outside diameter to be drilled. Possibly a lead is required in the workpiece to easily allow a dowel pin to easily slide into the hole. The drawing shows a picture of a common spot drill. Notice that the spot drill has a ninety degree point angle which can easily form a 45 degree chamfer for the hole being machined.



90 degree spot drill

The depth of the spot drilled hole determines the chamfer size, and is very easy to calculate. Simply divide the desired chamfer diameter by two to come up with the depth. The next drawing shows this.

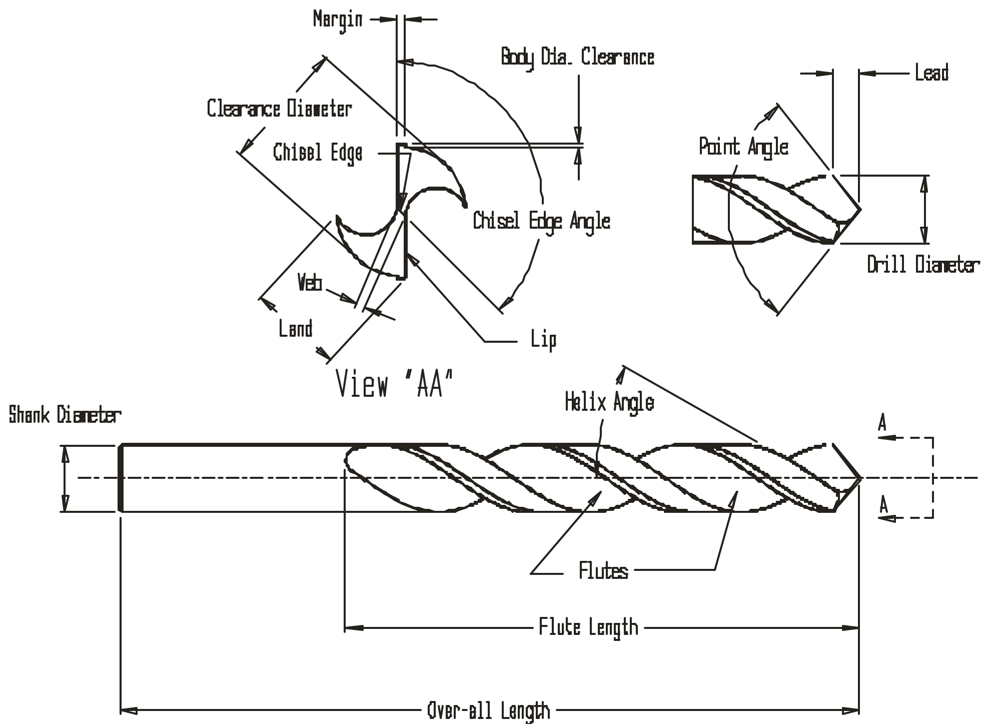


Depth of spot drilled hole is easy to calculate for a 90 degree spot drill. Simply divide the diameter of the chamfer by two.

For example, say you are going to drill the hole to a diameter of 0.250 (1/4) in. You want a 0.0312 (1/32) by 45 degree chamfer around the top of the hole. In this case, the desired chamfer diameter is 0.3124 (5/16) in (0.25 plus 0.0312 plus 0.0312). The depth required for the spot drill in this case is 0.1607 (0.3214 divided by 2).

1.1.1.2. Twist drills

There are many different styles of drilling tools. The most common is a twist drill. The next drawing shows a twist drill with the major notations shown.



Twist drill with terms and specifications.

For most woodworking applications, the twist drill is plunged to its final depth in one pass. The drill is then quickly retracted from the hole.

Allowing for the Drill Point

When most design engineers specify the depth of a hole, they usually mean for the hole to be machined with the specified diameter to the depth given. This means if a twist drill (or any drill with a point angle) is used, the programmer *must add* the drill point amount (called the *lead* of the drill) to the depth of the hole specified in the program. While there are times when the design engineer will specify the hole depth to the very point of the drill, the drill point must usually be taken into consideration when a hole is machined.

A hole which does not pass completely through the surface being machined is called a *blind hole*. A hole which passes completely through the surface being machined is called a *through hole*.

For a standard 118 degree point angle twist drill, the amount of drill lead is easy to calculate. Simply multiply the diameter of the drill times the constant value 0.30 to come up with the drill lead. For example, a 0.50 (1/2) in diameter twist drill has a 0.15 in lead (0.3 times 0.5). In this case, the value 0.15 must be added to the hole depth specified on the drawing to come up with the hole depth for the program.

Calculating the depth of through holes

If machining a hole through a surface with a twist drill, of course the drill lead must be added to the workpiece thickness. Additionally, you must add a small amount of clearance which forces the drill to truly break through the surface. Normally 0.030 (about 1/32) in is usually sufficient. If you do not add this small amount to your hole depth, it is likely that the hole will not be completely machined through the surface. This is especially likely if the material is pliable, having the tendency to push away from the cutting edges of the drill. For example, if machining through a 1.0 in thick workpiece with a 0.500 diameter drill, you should program the depth of the hole as at least 1.180 in (1.0 plus 0.15 lead plus 0.030).

Peck drilling to break chips

In some materials, the motions described above (feed to depth, retract) for drilling will cause a long stringy chip to be formed as the drill plunges the hole to its final depth. Many kinds of plastics and metals are notorious for this kind of stringy chip. The long chip will be whipped around the drill, and grow longer and longer. If left to grow, this long skinny chip will eventually break and be thrown away from the drill. If the protective guarding around the machine is inadequate, it is quite possible that the chip will be thrown at the operator, possibly causing injury.

One way to solve the stringy chip problem is to force the chip to break at manageable lengths as the hole is being drilled. In this case, *peck drilling* can be done to break chips. For example, the drill can be plunged into the hole a short distance (say 0.125 [1/8] in). Then the drill can be retracted a very small amount (about 0.005 in). It is during this small retract motion that the chip is forced to break. This plunge and small retract can be repeated for the entire hole depth. Since this kind of motion would be somewhat difficult (and quite lengthy) to program, most control manufacturers offer a special peck drilling *canned cycle* for the purpose of breaking chips (discussed in key concept number six).

Peck drilling for deep holes

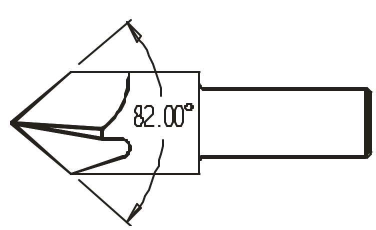
The *flutes* of a twist drill limit the drill's maximum drilling depth (see the previous drawing of a twist drill). However, you must know that most twist drills can not machine to this flute-length-depth in one pass. If this is attempted, the flutes of the twist drill will pack with chips, and these chips will eventually bind up between the drill and the workpiece. If drilling continues after this binding, the workpiece may be damaged and the drill will eventually break. For this reason, when deep holes must be machined, the drill must peck into the hole a specified depth, then retract completely out of the hole to clear the chips. Then, if the hole must be machined deeper, the drill can be sent back into the hole to within a small clearance distance from where it left off. The hole can then be machined to a greater depth.

For most materials, this maximum peck depth can be calculated by multiplying the diameter of the drill times three. That is, a twist drill has the ability to machine to a depth of about three times its diameter without fear of the chips packing up. If the required depth of hole is deeper than three times the drill depth, you should use peck drilling techniques to clear the chips.

For example, say you must machine a 0.500 diameter hole to a depth of 2.5 in. Three times 0.500 in is 1.5 in. In this case, you will first peck to a 1.5 in depth, then retract the drill to clear chips. You will then send the drill back into the hole to a clearance position just above where the drill left off (into the hole by 1.4 in will work nicely for this example). Finally, you will command that the balance of the hole be drilled. Since deep hole peck drilling is often necessary, and since it can be cumbersome to program as just described, most machining center controls have a *deep hole peck drilling canned cycle* for clearing chips in this manner.

1.1.1.3. Countersinking

The purpose for countersinking is to provide clearance for flat head screws. Countersinking is done *after* the hole is machined (usually by drilling). A countersink is not designed to machine a hole into solid material. The drawing shows a countersink.

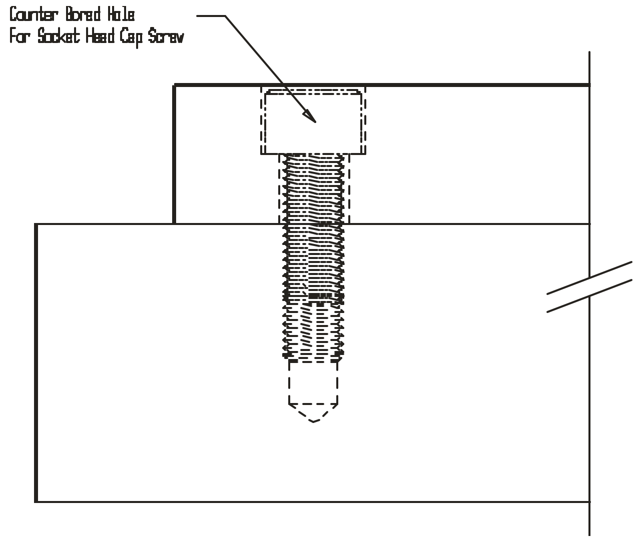


Countersink used to make clearance for flat head screws

Notice the 82 degree included angle of the countersink. It makes calculating the programmed depth harder than it is for a spot drill. To calculate the desired depth, divide the *radius* of the hole to be countersunk by the tangent of forty-one degrees (0.8693). For example, say you wish to countersink a hole to a diameter of 0.5 (1/2) inch. In this case, divide 0.25 by 0.8693 and the programmed depth will be 0.2876 inch.

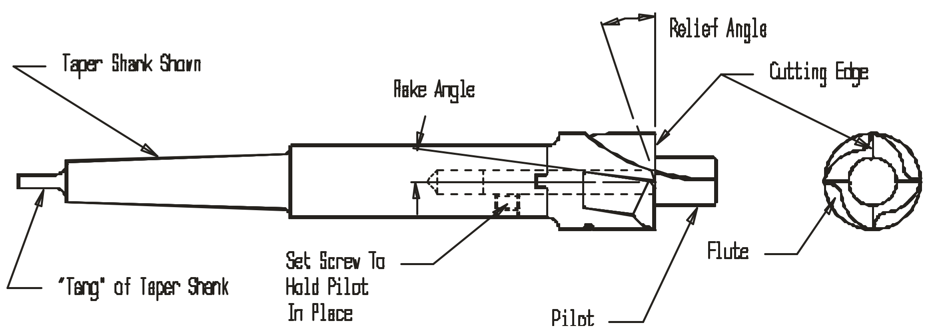
1.1.1.4. Counterboring tools

Counterboring is performed *after* a hole has been machined (usually by drilling) to open the hole to a larger diameter. The bottom of the hole machined by the counterboring tool is flat. The purpose for counterboring is usually to relieve a surface for the head of a screw or bolt. The drawing shows an application requiring a counterbored hole.



Application for counterboring

A true counterbore has a pilot that locates in the previously machined hole and keeps the counterbore from wandering as the hole is machined. The next drawing shows a drawing of a counterbore.



A true counterbore

If a true counterbore is used for the counterboring operation, the pilot is actually sent into the hole until the cutting edge is within the clearance position of the work surface. You must be concerned with the pilot's position after the hole is machined if several holes must be counter-bored. We'll show an easy way to clear the pilot between holes when we discuss the *counterboring canned cycle* in key concept number six.

Keep in mind that most companies use either a straight flute router bit or an *end mill* for counterboring purposes to minimize the number of tools required. (In fact, many experienced programmers that have never used a true counterboring tool.) All straight, flat bottom router bits and end mills allow machining in a plunging direction. However, if the cutting tool is not *center cutting* (it can not machine into solid material), you must be careful that the previously machined hole is larger than the hole in the center of the router bit or end mill. A router bit or end mill does not require a pilot so there is no need to clear the pilot between holes.

1.1.2. Routing

The primary application for a CNC router is, of course, to perform routing operations. In this section, we show many considerations related to routing, including feed direction, feeds and speeds, and router bits available.

1.1.2.1. Cutting conditions for routing

The process of routing involves rotating a tool (the router bit) at an appropriate speed (commonly specified in rpm) and driving it into the workpiece in a controlled manner at an appropriate traverse rate (feedrate). During contact, workpiece material is shaved from the workpiece in small amounts by the cutting edge/s of the router bit. *Cutting conditions* like rotation speed, feedrate, depth-of-cut, feed direction, workpiece material, and router bit material determine the quality of the routing operation.

1. Bit rotation speed

Rotation speed is specified in rpm on CNC routers. The rpm selected must be appropriate for the workpiece material being routed, the router bit material (high speed steel, carbide, etc.) and diameter of the router bit. CNC routers have very fast spindle speeds. It is not uncommon for current model CNC routers to allow speeds in excess of 30,000 rpm. Most router bit manufacturers specify their speed recommendations in *surface feet per minute*, based upon the material you are routing. This means you'll have to calculate the rpm for the router bit by applying this formula.

Rpm = 3.83 times SFM divided by bit diameter

If, for example, the router bit manufacturer recommends that you run a 0.5 (1/2) in bit at 1,000 sfm in a given workpiece material, multiply 3.82 times 1,000 (result is 3,820) and divide this result by 0.5 (rpm required is 7,640 rpm.

Many router bits have several diameters to allow form routing (any molding router bit will have more than one cutting diameter). When calculating rpm for a multiple-diameter router bit, *always base the rpm calculation on the bit's largest diameter.* While the calculated rpm may be somewhat slow for the smaller diameters of the bit, at least the bit will not overheat.

2. Feedrate

Feedrate is the traverse rate as the router bit moves along its cut. Most CNC routers require feedrate specification in *per minute* fashion (either inches or millimeters per minute). Like spindle speed, most router bit manufacturers will supply recommended feedrates in their technical specifications. This feedrate is usually specified in inches per tooth (flute, or insert) or in inches per revolution. Here are some helpful formulae to calculate feedrate.

Ipr = ipt times number of flutes or inserts

Ipm = ipr times rpm

Say for example, the router bit manufacturer recommends a feedrate of 0.004 ipt for the router bit used in the previous rpm calculation. The bit has two flutes or inserts, so the ipr feedrate for this bit will be 0.008 inch. Based upon the previously calculated rpm of 7,640, this bit should be run at a feedrate of 61.12 inches per minute.

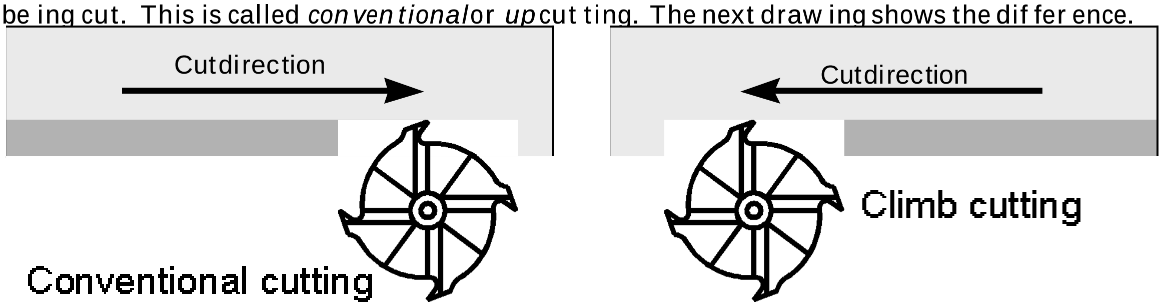
Note that there could be a spindle horsepower constraint related to your calculated feedrate. While the correct feedrate will be 61.12 ipm, we're assuming the machine spindle has adequate horsepower to drive the router bit through its motions at optimum cutting conditions. For lighter duty machines, youmay have to compromise the cutting conditions to avoid stalling the spindle or axis drives for larger tools or when taking heavier cuts.

3. Axial and radial depth-of-cut

Axial depth-of-cut is how deep the router bit is plunged into the workpiece along its length. Radial depth-of-cut is how deep the router bit is cutting over its diameter. If routing a slot, for example, the radial depth of cut will be the router bit diameter. The deeper these depths of cut, the more material will be removed from the workpiece during the routing pass. Most speed and feed recommendations will take into consideration these depths of cut.

4. Feed direction

When machining on the periphery of the router bit (side cutting), you can feed the bit in one of two directions relative to the bit's rotation direction. In one direction, the router bit will be pulling itself along. This is called *climb* or *down* cutting. In the other, the router bit will be pushing itself away from the material being cut. This is called *conventional* or *up* cutting. The next drawing shows the difference.



Climb versus conventional cutting.

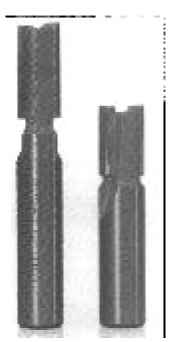
Though there are a few exceptions, woodworkers using hand held routers are usually very careful *never* to climb cut. Since the router bit has the tendency to pull itself along the cut, the hand held router will be difficult to control. It may be pulled right out of the woodworker's hands. But CNC routers can usually hold the router head much more securely than a person can. It is not only possible to climb cut on a CNC router, it is often desirable, given the better finish that can be achieved by climb cutting.

1.1.2.2. Types of router bits

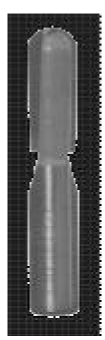
Here we show a few common router bits. There is a wide variety of standard bits available for routing almost any conceivable shape. Additionally, special bits can be designed and manufactured for applications having no standard bit.

Note once again that there are many excellent texts available that describe router bits in much greater detail than we do. Again, our intention is only to acquaint you with those types of tools we assume you understand throughout the balance of this course.

1. Surface or groove forming bits



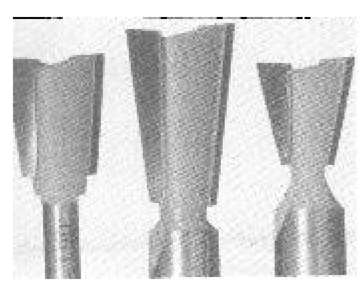
Straight nose bit



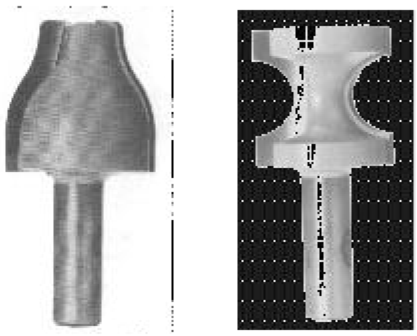
Round nose bit

Angle nose bit

Tee slot bit



Dovetail bits



Edge forming bits

One thing you may have noticed is that none of these bits have pilots (solid or ball bearing). If you have experience with hand held routers, you know that pilots are often required to help you keep the router bit following the correct path. But since CNC routers have very precise (and well supported) motion control, pilots are not necessary.

Also notice that several of these bits have multiple diameters. When it comes to rpm calculation, again, you should base your rpm choice on the bit's *largest diameter*.

2. Carbide insert router bits

More and more standard bits are available with carbide inserts. Instead of having to replace the entire bit when it gets dull, the operator or tool setter will simply replace the carbide inserts within the bits. This minimizes the amount of time it takes to replace worn cutting tools.

1.2. Machine configurations

This section explains the basic make-up of a CNC router. Most beginners tend to be a little intimidated when they see a CNC router in operation for the first time. Admittedly, there will be a number of new functions to learn. The first point to make is that you *must not* let the machine intimidate you. As you go along in this course, you will find that a CNC router is very logical and is almost easy to understand with proper instruction. If you have previous machining experience with *manually* operated equipment (like hand held routers and drill presses), you will find that most of what you already know about the basic machining practice of woodworking will directly apply to CNC routers.

You can think of any CNC machine as being nothing more than the standard type of equipment it is replacing (hand-held router, drill press, etc.) with a very sophisticated control added. Instead of activating things manually by handwheels and manual labor, you will be preparing a *program* that *tells* the machine what to do. Virtually anything that needs to be done on a CNC router can be activated through a program.

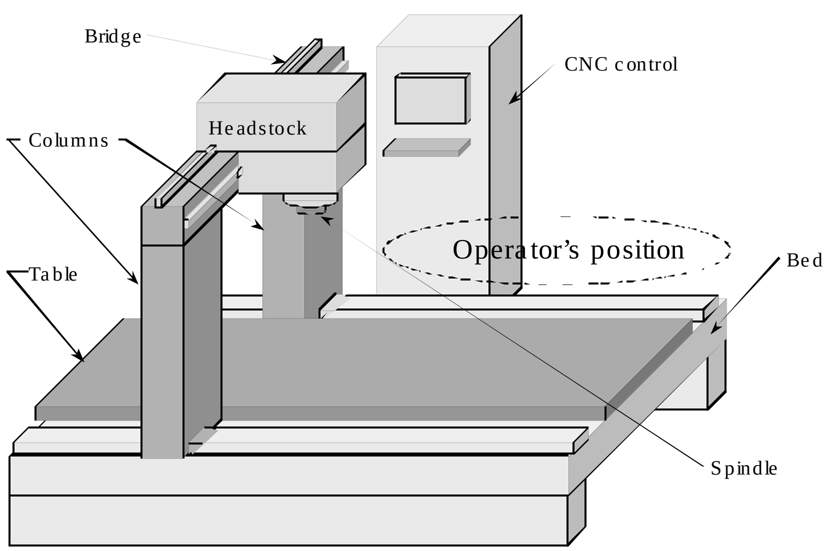
1.2.1. CNC routers versus CNC woodworking machining centers

There are two basic types of CNC routers that we will be addressing in this course. They are the CNC router and the CNC woodworking machining center. For the purpose of this course, about the only difference is that the CNC machining center has an automatic tool changing (ATC) system while the CNC router does not. Though there are exceptions, CNC routers tend to be smaller (some are even table top sized) and less expensive. Because they are so simple to work with, many companies purchase CNC routers (as opposed to machining centers) as their first CNC machine.

There may be some other differences between CNC routers and machining centers having to do with programmable features. Some lower cost CNC routers, for example, may not have programmable spindles. In addition to manually loading the first tool, you may also have to activate the spindle manually (turn it on in the desired direction at the proper speed) before you can activate the CNC program. With *full blown* CNC machining centers, *all* machine functions will be programmable, minimizing the amount of operator intervention during the CNC cycle. Many of the examples shown in this course assume you'll be working with a full blown CNC machining center.

1.2.1.1. Basic components

Most CNC routers and machining centers incorporate a *bridge* design. The workpiece will be held in a stationary position on the *machine table* while the cutting tools move around it to perform machining operations. This drawing shows the basic components for the most common style of CNC router and machining center.

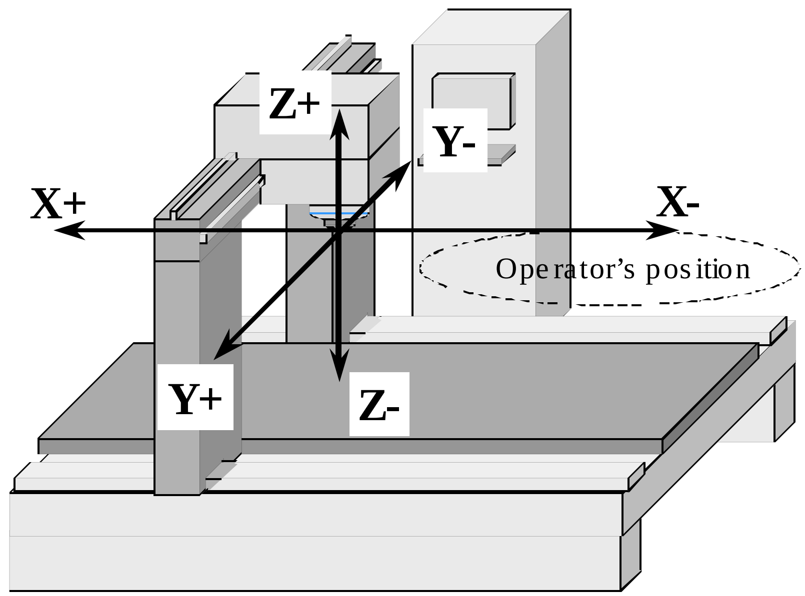


Components for a CNC router or machining center

While designs will vary from one router manufacturer to another, notice that the headstock and spindle are held atop the bridge. The workpiece is mounted, in one fashion or another to the machine table. While workpiece securing devices vary, a common work holding method is a vacuum-suction system. With this workholding method, the work holding table incorporates a vacuum system to securely hold workpieces during machining. Once the workpiece is placed in position, the vacuum system is activated during workpiece loading from the operator's panel.

Most CNC routers incorporate another style of vacuum system to suck the chips from the work area. Though it may be programmable, this device is also commonly activated from the operator's panel (with some machines, this vacuum system is automatically activated when the spindle is turned on).

The cutting tool held in the machine spindle will move in at least three directions. Each motion direction is called an *axis*. The three most basic axes are *linear*, allowing the cutting tool to move along a perfectly straight line. The next drawing shows the naming and polarity of each motion direction.



Directions of motion for CNC routers

Though some CNC router manufacturers stray from this standard, the longest axis is usually specified as the *X axis*. Note that the operator stands on the *other* side of the table, not on the side from which we're viewing. From the operator's position, the X axis is usually the left/right direction of motion. The Y axis is the fore/aft motion direction (toward and away from the operator). And the Z axis is the spindle motion direction toward and away from the table top (up/down). Most CNC routers incorporate a *quill* within the spindle to form the Z axis.

At first glance the axis polarities for X and Y may appear to be reversed. But notice once again the position from which we are viewing this machine. The operator actually stands on the *other side* of the table. Tool motion to the operator's right is the X plus direction. Tool motion away from the operator is Y plus motion.

One of the first things a CNC router programmer must learn are the motion directions (axes) and their polarity for the machine/s they will be programming. While most CNC routers and machining centers adhere to the methods shown in the previous drawings, you must confirm them for the machine/s with which you will be working. You may find that the X and Y axes are reversed. You may also find their polarities seem to be reversed. But regardless of how your axes are arranged, once you can view the machine from the proper perspective, you can easily orient the drawings we show in this course to the machine/s with which you will be working.

1.2.2. Other programmable feature of CNC routers and machining centers

As mentioned, today's *full blown* woodworking machining centers allow the programmer to control just about any function required through programmed commands. But regardless of how many programmable functions your CNC routers and machining centers have, it is very important that you undersand *what* is programmable. We can not be specific for your own company's machines. You'll have to check with an experienced person in your company or school, or look in your machine tool builder's programming manual to determine which machine functions are programmable.

Here we list the things that the programmer can usually control within the program and give a cursory view of how each function is controlled. Our intention here is *not* to teach programming commands (yet). It is to simply make you aware of the kinds of things a programmer can control through a program. If you'll be working with a CNC router (no automatic tool changer), you'll need to confirm which, if any, of these features is programmable.

1.2.2.1. Tool changing

Most CNC woodworking machining centers use a T word to specify which tool should be placed into the *ready* or *waiting* position (just about to be placed in the machine's spindle) and an M06 to actually cause the tool change. For example, the command

```
T07 M06
```

will place tool number seven into the spindle. Most machines require that the Z axis be in a special position to allow the tool change to occur. The tool change position for most CNC machining centers is the machine's Z axis *zero return* position. We'll describe the zero return position a little later in this key concept.

1.2.2.2. Spindle speed

A programmer can control precisely how fast the spindle rotates in one RPM increments. An S word is used for this purpose. If the programmer wishes 2,350 RPM, the word S2350 is commanded.

1.2.2.3. Spindle direction

A programmer can also control which direction the spindle rotates, forward or reverse. The forward direction is commonly used for right hand tooling and the reverse direction is used for left hand tooling. Two *M codes* control this function. An M03 turns the spindle on in the forward direction. M04 turns the spindle on in a reverse direction. Since almost all cutting tools used with CNC routers are right hand, you'll almost always be using M03 (forward direction) to start the spindle. M05 turns the spindle off.

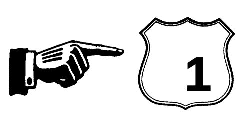
1.2.2.4. Feedrate

A programmer can control the motion rate for any machining operation. This is done with an *F word*. The F word specifies feedrate in *per minute* mode. If you wish to work in the inch measurement system, this feedrate will be in inches per minute. If you work in metric, the feedrate will be in millimeters per minute. A feedrate of 3.5 inches per minute is programmed as F3.5.

1.2.2.5. What else might be programmable?

While this course will acquaint you with the most common programmable functions of CNC routers and machining centers, you must be prepared for more. Other features that *may* be equipped on your own machines include rotary axes (to allow the tool or workpiece to be rotated during machining operations), air blowing systems (to clear chips from the work area), and a variety of other application based features. If you have any of these features, you must reference your machine tool builder's programming manual to learn how these special features are programmed.

STOP!! Do practice exercise number:



Lesson 2

1.3. General flow of the programming process

Though it is not directly related to the CNC router or machining center itself, we want to introduce how the typical company using CNC equipment would process a job to be done on a CNC router. This should help you understand the general method by which CNC programs are processed and should also help you grasp the specific task of programming. In general, it will help you see the *big picture* before diving into learning the more specific task of programming.

Here we give a generic example of how a typical shop would handle the CNC programming task. By no means will this flow be the same for all shops. Smaller shops tend to have one or two people handling all steps in this flow. Larger manufacturing companies will usually break these steps up to be handled by several people, possibly in different departments.

1.3.1. Decision is made as to which CNC machine to use

If the company has more than one CNC router or machining center, there may be some question as to which one to use. Based on the required number of workpieces, the size of the workpiece, the accuracy required of the workpiece, the material to be machined, and the shop loading on any one CNC router (among other things), a decision is made as to which CNC machine to use.

1.3.2. The machining process is developed

If more than one operation must be performed by the CNC machine, someone with woodworking background (perhaps the programmer) must come up with a sequence of operations to be used to machine the part. The program will follow the same sequence.

1.3.3. Tooling is ordered and checked

Based on a previously developed process, the required (workholding and cutting) tooling is obtained. It is helpful early on in the process to determine what tooling, if any, must be ordered.

1.3.4. The program is developed

In this step, the programmer *codes* the program into a language that the CNC machine can understand. This step is, of course, one of the main focuses of this course. We present six key concepts to help you understand this step.

1.3.5. Setup documentation is made

Part of the programmer's responsibility is to make it clear as to how the setup is to be made at the machine. Drawings can be made to describe the work holding setup. If multiple tools are to be used in the program, some kind of tool list including the machine's magazine station numbers to be used for each tool. The more often a job is repeated, the more important it is to document how the setup must be made.

1.3.6. Program is loaded into the CNC control's memory

Once the program is prepared, there are two common methods used to load it into the control's memory. One way is for someone to physically type it through the keyboard and display screen of the control panel. Generally speaking, this is a rather cumbersome way to get the program into the control's memory. For one thing, the keyboard of the control panel is quite difficult with which to work. The keyboard is not oriented in a logical way (most are *not* like the keyboard of a typewriter) and usually the panel itself is mounted vertically, resulting in the person typing the program experiencing fatigue. In many cases, the machine will sit idle while the program is being typed. Even a relatively short program could take thirty minutes or more to enter. A CNC machine makes a *very* expensive typewriter!

Another more popular way to enter programs into the control's memory is to use some outside device for typing programs. A desktop computer can be used for the this purpose. The software used for this computer application resembles a common *word processor*. In fact, most word processors can be used for the purpose of typing CNC programs. Once the program is entered through the computer's keyboard, it can be saved on the computer's floppy disk or hard drive. When the program is needed at the machine, the program can be quickly transmitted to the machine from the computer. This transmission takes place almost instantaneously, even for lengthy programs. The computer makes it much easier for the person typing the program. They can sit in a much more comfortable environment to type the program. Most importantly, the program can be typed while the machine continues in production for the current production run, meaning almost no machine time is wasted while the program is loaded.

1.3.7. The set up is made

Before the program can be run, the workholding set up must be made. Using the set-up instructions (commonly prepared by the programmer), the set up person makes the set up. Tooling is assembled and loaded into the proper locations. Measurements must also be made to determine the position of program zero and the length of each tool. And, in general, any function related to the set-up must be done.

1.3.8. The program is cautiously verified

It is *very* rare that a new program requires no modification at the machine. Even if the programmer does a good job programming the motion required, there will almost always be some optimizing necessary to reduce the cycle time to the minimum. During key concept number ten we discuss program verification in great detail.

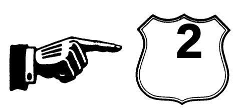
1.3.9. Production is run

At this point the machine is turned loose to run production. While the programmer's job could be considered finished at this point, there may be some long term problems that do not present themselves until several workpieces are run. For example, tool wear may be excessive. Tools may have to be replaced more often than the company would like. In this case, the speeds, feeds, and depths-of-cut in the program may have to be adjusted.

1.3.10. Corrected version of the program is stored for future use

At least some workpieces run on CNC machines are run on a regular basis, especially in manufacturing companies that produce a product. At some future date it will probably be necessary to run the workpiece again. If changes are necessary during program verification, it will be necessary to transmit the *corrected* CNC program back to the program storage device for future use. If the company must run the same workpieces again at some future date, the corrected version of the program must be kept. If this step is not done, of course, the program will have to be verified a second time.

STOP!! Do practice exercise number:



This page can be used for notes.

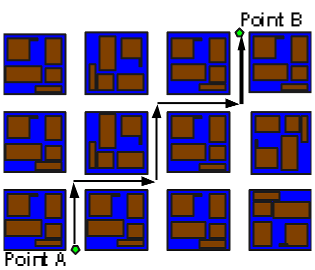
Lesson 3

1.4. Visualizing the execution of a CNC program

A *CNC programmer must have the ability to visualize the movements a CNC machine makes.* This is true of *all* forms of CNC equipment, including CNC routers and machining centers. The better a person can visualize what the CNC router will be doing, the easier it will be to prepare a correct CNC program. To stress this point, think of how a manual drill press operator uses their machine.

When an drill press operator prepares to machine a workpiece, he or she has all related components of the job right in front of them. The machine, cutting tools, workholding setup, and print are ready for immediate use. It is highly unlikely that the operator will make silly mistakes like forgetting to start the spindle before trying to drill a hole in the workpiece.

On the other hand, a CNC router or machining center programmer will be writing the program with only the workpiece and tooling drawings for reference. No tooling - no machine - no workholding setup will be in front of them. For this reason, the programmer *must* be able to visualize just exactly what will happen during the execution of the program - and this can sometimes be difficult, since visualization must take place in the programmer's mind. *A beginning programmer will be prone to forget certain things* - sometimes very basic things (like turning the spindle on prior to machining the workpiece).



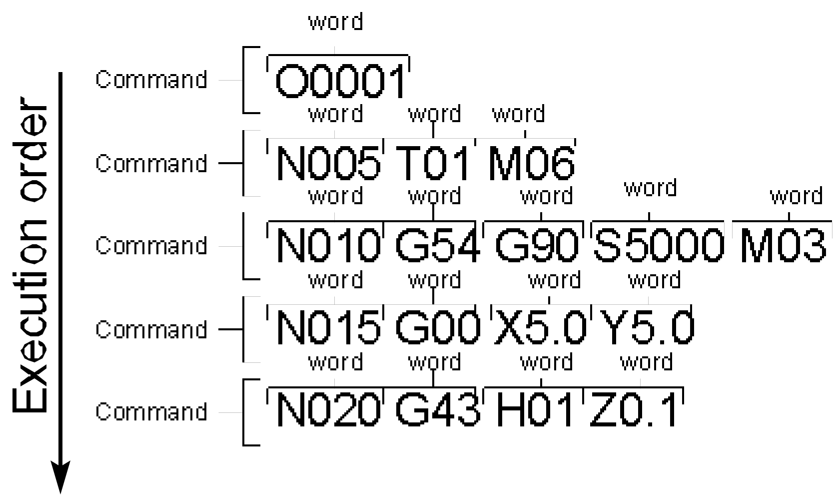
This can be likened to developing a set of travel instructions. In order to be able to write down a set of travel instructions, you must be able to visualize the path from the start point to the end point. Only with this path *visualized*, can you write down a sequential step-by-step set of instructions to get a person from point A to point B. If you make a mistake in your set of travel instructions, the person following your instructions will get lost.

In similar fashion, a CNC programmer must be able to prepare a set of step-by-step instructions for a CNC router to follow. And visualizing what the machine will be doing is every bit as important, since the machine will follow the commands you give *exactly as you give them*. Mistakes are every bit as serious. If you make a mistake, the CNC machine will not perform as desired!

In this section, we intend to acquaint you with those things a programmer *must* be able to visualize. We will also show the first (elementary) program example to stress the points being made.

1.4.1. Program make-up

Like the English *sentences* making up your set of travel instructions, a CNC program is made up of *commands* (also called *blocks* or *lines*). Within each command are *words*. Each word within a command is made up of a letter address (N, X, Z, T, etc.) and a numerical value. The next graphic shows an example portion of a CNC program stressing program make-up.



Program make-up and execution order.

1.4.1.1. Method of program execution

You can also compare a CNC program to giving a set of step-by-step assembly instructions. For example, say you have just purchased a bookcase that requires assembly. The instructions you receive will be in sequential order. You will perform step number one before proceeding to step number two. Each step will include a sentence or paragraph explaining what it is you are supposed to do at the current time. As you follow each step, performing the given procedure, you are one step closer to finishing. You may even have a checkbox to mark as you finish each step along the way to completion.

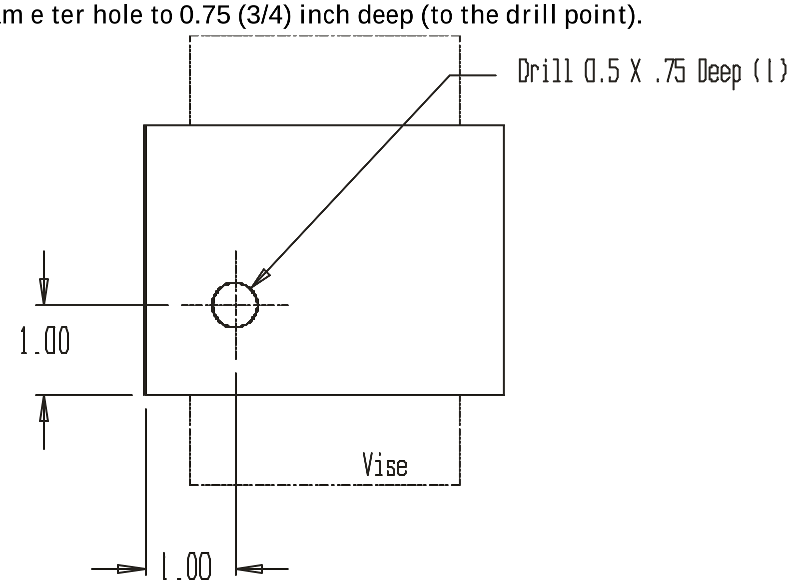
In similar fashion, a CNC program will be executed in sequential order. The CNC control will read, interpret, and execute the first command in the program. Then it will go on to the next command. Read, interpret, execute. The control will continue this process for the balance of the program.

Keep in mind that the CNC control will execute each command *explicitly*. Compare this to the set of instructions for assembling a bookcase. In the set of assembly instructions, the manufacturer may be rather vague as to what it is you are supposed to do in a given step. They may assume certain things. They may assume, for example, that you have a screwdriver and that you know how to use it. This set of instructions may be so vague that it is open to interpretation. A CNC control, by comparison, will make *no* assumptions. Each command will be very *explicit*. The CNC control will execute each command just as it is written. And any given CNC command will have but one resultant machine action.

An example of program execution

To stress the point, we look at a simple example. We first show the steps a drill press operator would perform in order to machine a workpiece. Then we will show the equivalent CNC program. Our intention is to show you the things you must be able to visualize before you can write CNC programs.

The next drawing shows the print for this extremely simple operation. In this case, we are simply drilling a 0.5 (1/2) inch diameter hole to 0.75 (3/4) inch deep (to the drill point).



Drawing for example illustrating program execution flow

First let's look at the procedure a drill press operator would perform. Note that we assume that the workpiece is already in a vise and that the drill is already in a chuck in the spindle.

Drill press machine procedure:

1) Turn spindle on CW at 600 RPM
2) Move the drill up close to position and ready to drill
3) At desired feedrate, drill the 0.500 hole
4) Retract the drill from the hole
5) Move tool away and turn off the spindle

Admittedly, this example is extremely simple. But keep in mind that we are stressing the *sequential* order by which an operator machines the workpiece and the *visualization of these steps that is mandatory in order to write a CNC program to drill this hole.*

CNC program:

Now here is a CNC program to drill the 0.5 (1/2) inch diameter hole on a CNC router. Again, we assume that the drill is in the spindle and that the workpiece is held in a vise on the table when this program begins. Notice that we document each command with a message in parentheses in order to help you see what is happening during each command. Note that you can include messages within parentheses in your own programs (though in actual programs they must be all in capital letters) for documentation purposes.

O0001 (Program number)

N005 G54 G90 S600 M03 (Assign program zero, select absolute mode, turn spindle on CW at 600 RPM)

N010 G00 X1.0 Y1.0 (Move the tool into position in X and Y)

N015 G43 H01 Z0.1 (Rapid up to workpiece, instating tool length compensation)

N020 G01 Z-0.750 F3.5 (Drill hole at 3.5 IPM)

N025 G00 Z0.1 (Rapid out of the hole, turn off coolant)

N030 G91 G28 X0 Y0 Z0 (Rapid to the machine's reference position)

N035 M30 (End of program, this command also turns off spindle)

Though the actual commands in this program may not make much sense yet, the messages within parentheses should nicely clarify what is happening in each command. Again, our intention is *not* to teach the programming words being used (yet). Our intention *is* to stress the importance of your ability to visualize the operation/s being performed. In this case, you must be able to *see* the drill machining the hole (in your mind) before this program can be written. We're also stressing the *sequential order* by which the program will be executed. The control will execute line number N005 before moving on to line number N010. Then line number N015. And so on —until the entire program is activated.

Also, you should now be able to see how your previous woodworking experience is going to help. If you can visualize how you want tools to move, as shown in the drill press example, you are well on your way to learning how to program these movements for a CNC router. Indeed, if you *cannot* visualize how you would want a tool to move as it performs a machining operation, you will not be able to write the CNC program. For those machining operations with which you are not familiar, you will (at the very least) need the help of an experienced woodworker.

1.4.1.2. A note about decimal point programming

As you have seen in the previous example program, many CNC words allow a decimal point to be given within a numerical value. *All* current controls allow decimal point programming in those words which require real numbers (but not within words that simply require only integer [whole] values).

Beginning programmers have the tendency to leave out required decimal points, especially when programming integer values in words that allow a decimal point. It is *mandatory* that you include a decimal point within each word that allows a decimal point. If you do not, some very strange things can happen. Here's why.

Older controls do not allow a decimal point in any CNC word. These controls require a *fixed format* for all real numbers needed in the program. Trailing zeros are required for these older controls to *imply* where the decimal point should be placed. For example, an X motion word of 5.0 in would be specified as "X50000" if decimal point programming is not allowed.

Newer controls are upward compatible. This means programs written for older controls can still run in current controls. If a current CNC control sees a word that allows a decimal point without a decimal point included in the word, it will simply place the decimal point automatically using the fixed format.

Here's an example. Say the programmer intends to specify an X word of 3.0 in. The correct way to designate this movement is "X3.0" But say the programmer makes a mistake. He or she incorrectly programs

"X3", leaving out the decimal point. In this case, the control will incorrectly interpret the X movement. Instead of taking this command as 3.0 inches, the control will place the decimal point *four places to the left of the right-most digit*. In this case, "X3" will be taken as X0.0003, *not* 3.0 inches.

Remember that beginning programmers tend to make *mistakes of omission*, meaning your tendency will be to omit the decimal point in CNC words that require it. Get in the habit of including a decimal point within those words that allow it. These letter address words include: F for feedrate, I, J, & K, for circular motion commands, R for radius, and X, Y & Z for axis movements. Words that do *not* allow a decimal point and must be programmed as integer values include: N, G, H, D, L, M, S, T, O, and P.

One way to ensure that you will not forget to program decimal points is to carry out whole numbers to the very first zero (after the decimal point) for those words that allow decimal points. For example, program X2.0 instead of X2. to eliminate the possibility of forgetting needed decimal points. In similar fashion, program the *leading* zero when designating decimal portions of a whole number. For example, program X0.375 instead of X.375.

Note that almost all CNC controls do require that for values under one, you program the *decimal equivalent*. Though many prints show values with fractional dimensions, you must convert to decimal. The value along the X axis of 3-15/16, for example, must be programmed as X3.9375.

1.4.1.3. Other mistakes of omission

Knowing the mistakes beginners are prone to make may help you avoid them. Beginners tend to forget things in their programs. They forget to program a decimal point. They forget to turn on the spindle. They forget to drill a hole before counter-boring it. Much later, we will be showing you example program formats that should help you avoid these mistakes of omission. But you will have to concentrate very hard to avoiding being forgetful.

1.4.1.4. Modal words

You must know that many CNC words are *modal*. This means that the CNC word remains in effect until changed or canceled. In the previous example program, notice the G00 command in line N010. This happens to be a rapid motion command that tells the control to move into position in X and Y. Notice that the very next (line N015) makes the movement in Z to approach the workpiece. This movement will also occur as a rapid motion because G00 is *modal*. Modal words do not have to be repeated in every command.

1.4.1.5. Initialized words

You must also know that certain CNC words are initialized. This means the CNC control will assume these words to be instated at power-up. For example, most machines used in the US will come on in the inch mode (the instating word happens to be G20). If a company will be exclusively using the inch mode, they can depend upon the CNC control to be in this state at all times, meaning there is no need to actually include a G20 in the program.

1.4.1.6. Word order in a command

Note that with most CNC controls, the order by which CNC words appear in a command has no bearing upon how the command will be executed. For example, the command

N050 G00 X1.5 Y1.25

will be executed in exactly the same way as

N050 X1.5 G00 Y1.25

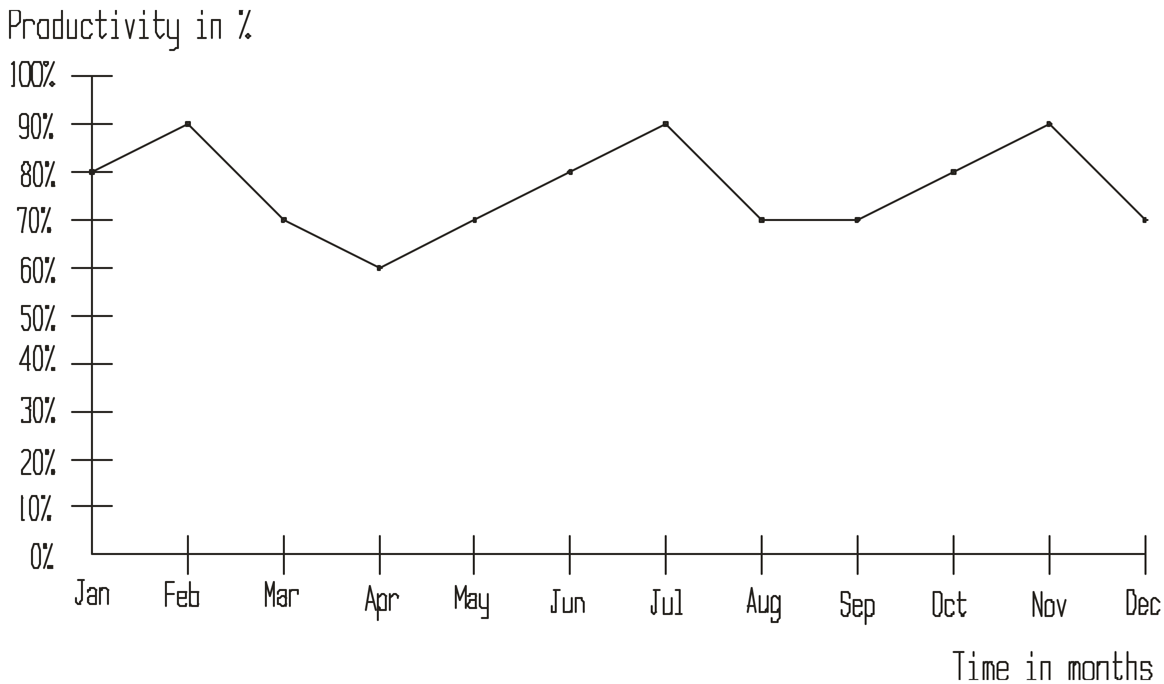
STOP!! Do practice exercise number:

Lesson 4

1.5. Understanding program zero and the rectangular coordinate system

The method by which positioning movement is commanded on CNC machines has been refined over the years. In the very early days of NC (well before *computer* numerical control), the NC programmer actually had to know how many rotations of an axis drive motor equated to an inch of linear movement. This, of course, made positioning movements very difficult to program. Today, thanks to the rectangular coordinate system, programmers need not concern themselves with the inner workings of each axis drive system in order to command positioning movements. With the rectangular coordinate system (also called the Cartesian coordinate system), positions, called *coordinates*, are specified with respect to a central origin point.

To help you understand the rectangular coordinate system, we give a simple analogy to making and using a graph. Everyone has had to make, or at least use, a graph. The next illustration shows an example.

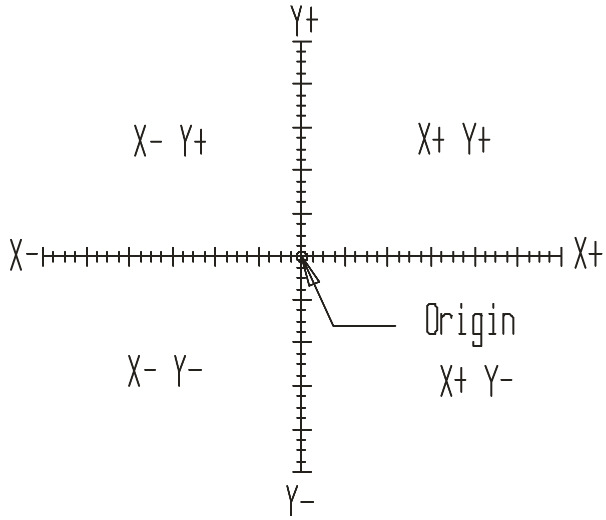


Graph example used to illustrate rectangular coordinate system.

This is a graph showing a company's productivity for last year. The horizontal base line represents time. The increment of time is specified in months. In this case, the *range* for the horizontal base line is one whole year from January through December. The vertical base line represents productivity. The increment for this base line is specified in ten percent increments and it ranges from 0% to 100% productivity.

In order for a person to make this graph, they would look up the productivity for January and plot a point along the vertical line corresponding to January and the percentage of productivity (80% in our case). This plotting of points would be repeated for every month of the year. Once all the points are plotted, a line or curve could be passed through each point to show anyone at a glance how the company did last year.

The graph analogy is amazingly similar to the rectangular coordinate system as it is used with CNC. For CNC routers and woodworking machining centers, the horizontal base line will represent the X axis. The vertical base line will represent the Y axis. (The Z axis is the direction is at a right angle to this page, toward and away from you.) The increments of each base line are given in linear measurement. For the inch mode (which we use throughout this course), each increment is given in inch. The smallest increment is 0.0001 inch, meaning each axis has a very fine grid. For the metric mode, each increment will be in millimeters. In the metric mode, the smallest increment is 0.001 mm (an even finer grid). The next drawing shows the XY plane coordinate system for a machining center.

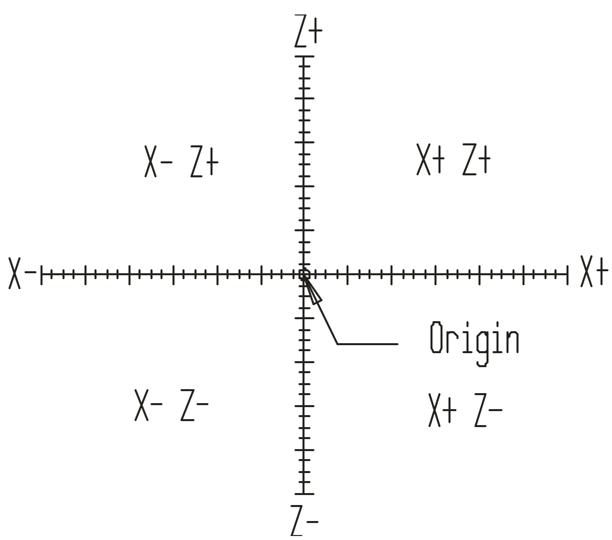


Rectangular coordinate system as related to the XY plane for CNC routers and machining centers.

Since most people in the United States are accustomed to the inch mode, we use it for all examples in this text. However, you should know that there is an accuracy advantage with the metric mode. The advantage has to do with the least input increment. As stated, in metric mode the least input increment is 0.001 mm, which is less than half of 0.0001 inch (0.001 mm is actually 0.000039 in). Think of it this way: A ten inch long linear axis has 100,000 positions in the inch mode. The same ten inch long linear axis has over 254,000 positions in the metric mode! This finer resolution allows the programming of more precise positions. Though most woodworking applications do not require the improved resolution the metric mode offers, it's nice to know the advantage exists.

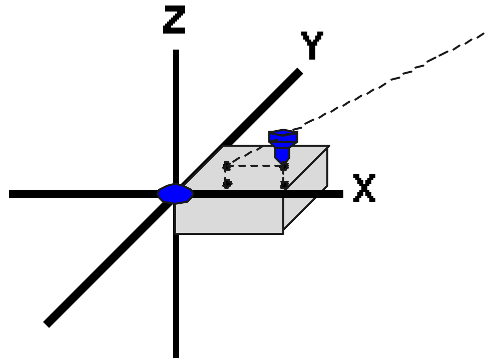
1.5.1. What about Z?

The previous drawing shows only two of the machining center's axes, X and Y. Keep in mind that the Z axis behaves in exactly the same manner as X and Y. The next drawing shows the X and Z plane (as if you are viewing the CNC router from the operator's position).



X and Z plane (as viewed the operator's position of a CNC router).

When taken all together, the X, Y, and Z give you a three dimensional grid. It is within this grid that you will be plotting positions (coordinates) that your tools will be passing through, as the next drawing depicts.



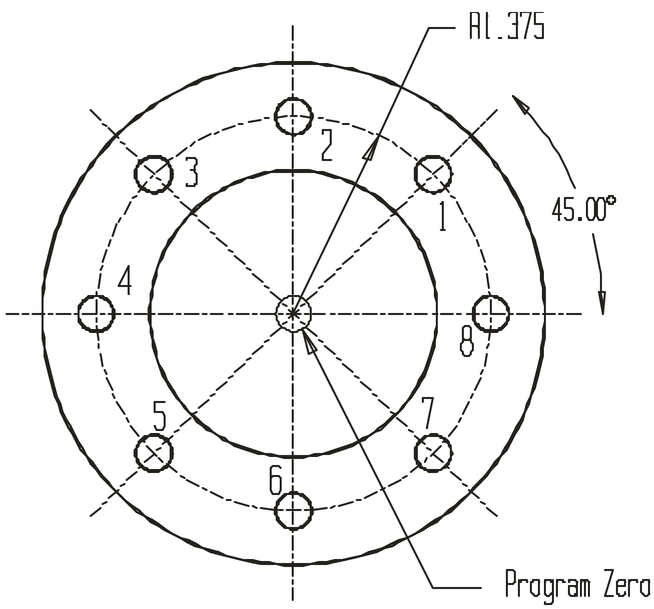
CNC routers allow tool positions to be plotted in a three dimensional system (X, Y, & Z)

1.5.2. Understanding plus and minus

The point where the base lines intersect is called the *origin*. In the graph example, notice that all points were plotted *after* January and above 0% productivity. The area up and to the right of the origin is called a *quadrant*. This particular quadrant is quadrant number one. The person creating the productivity graph intentionally planned for coordinates to fall in quadrant number one in order to make it easy to read the graph.

For CNC purposes, we call this origin the *program zero point*. There will be many times when coordinates used within the CNC program will not fall up and to the right of the program zero point (quadrant number one). Any coordinates that do not fall into quadrant number one require at least one negative position (ordinate). Note that the CNC control will automatically assume that a coordinate is plus unless a minus sign (-) is programmed within the coordinate word. This means you never have to program a plus sign (+).

The next drawing takes the example a little further and shows how a series of coordinates are determined. Notice that there is a series of holes around a circle that must be specified relative to the center of the circle. That is, the center of circle is the origin for the coordinate system. As you can see, any hole to the left of the origin requires a minus X coordinate. Any hole below the origin requires a minus Y coordinate. And remember, the CNC control will assume plus unless minus is specified.



#	X	Y
1	.9723	.9723
2	0	1.375
3	-.9723	.9723
4	-1.375	0
5	-.9723	-.9723
6	0	-1.375
7	.9723	-.9723
8	1.375	0

Drawing stresses plus and minus coordinates in X and Y axes.

1.5.3. **Program zero**

In the graph analogy, you know the graph has to start somewhere (where the base lines intersect). In our case, it started at January for the horizontal base line and 0% productivity for the vertical base line. When making graphs, this starting point for each base line is called the *origin* point.

For use with CNC, the origin point for each axis is most commonly called the *program zero point*. Also called *work zero*, *part origin*, and *zero point*, the program zero point allows the programmer to input all coordinates used in a program from a common and logical point. This point is usually the location on the print from which all (or the most) dimensions are taken. One of the main benefits of using the rectangular coordinate system to command positioning movements is that *many of the coordinates used within the program can be taken right from the print*.

In the graph, each base line has a *name, an increment, and a range*. For the horizontal base line, the name is time, the increment is given in one month intervals, and the range is from January through December (one year). For the vertical base line, the name is productivity, the increment is ten percent intervals, and the range is from zero to one hundred percent.

As stated earlier, each base line of the rectangular coordinate system represents a machine axis (X, Y, or Z). The increment for each base line is related to the measurement system. If working in the inch mode, most CNC routers will have a increment of 0.0001 inch (this is commonly called the machine's *least input increment* or *resolution*). In metric mode, the increment is usually 0.001 millimeters. The range of each base line is the overall length of travel for the axis it represents. If you have a CNC router with a seventy-two inch X axis, the range for your machine's X axis is seventy-two inches.

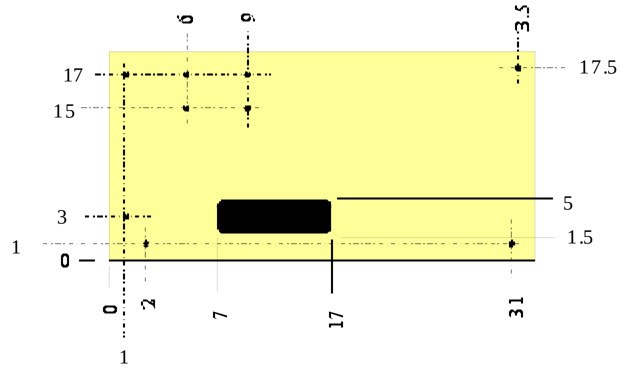
1.5.4. **Where to place your program zero point**

The placement of program zero is determined by the programmer. Program zero could be placed *anywhere*. As long as the coordinates in the program are specified from the program zero point, the machine will move properly. Though this is the case, the wise selection of a logical program zero point will make programming and setup easier.

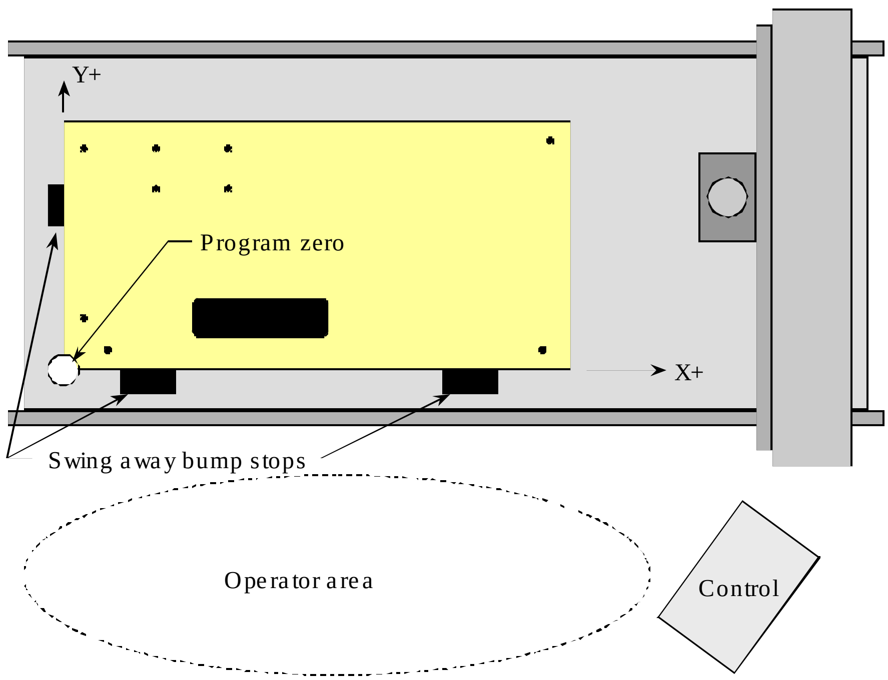
We recommend that you make your program zero point a location on the workpiece from which the dimensions in your program start. If the design engineer uses *datum surface dimensioning*, the program zero point selection will be easy. Just make it the datum surface on the workpiece for each axis. If the design engineer does not use datum surface dimensioning, they will almost always *begin* the dimensioning from one surface in each axis. This will be your program zero point. If you're still in doubt, ask yourself what surfaces you will be using to locate the workpiece in the machine for setup purposes. This surface in each axis usually corresponds to the program zero point.

Once the program zero point has been selected, *all* axis positions (coordinates) going into the program will be taken from this point. If you select the program zero point at the *datum surface* on the print for all axes, dimensions going into your program will be taken right from your print, minimizing the need to perform calculations in order to come up with program coordinates.

Much of the work done on CNC routers and especially CNC woodworking machining centers is done in rectangular *flat stock* (commonly cut from 4' x 8' plywood, fiberboard, etc.). Board thickness is commonly in the neighborhood of from 3/16 through one inch. The finished workpiece is also commonly square or rectangular in shape. For these workpieces, program zero will usually be one of the *corners*. Calculating coordinates will usually be pretty easy since most design engineers will specify all X and Y related dimensions from one corner. However, it will often be necessary to position the workpiece on the machine table in a way by which it can be easily located (bumped against X and Y program zero surfaces). The next drawing shows an example.

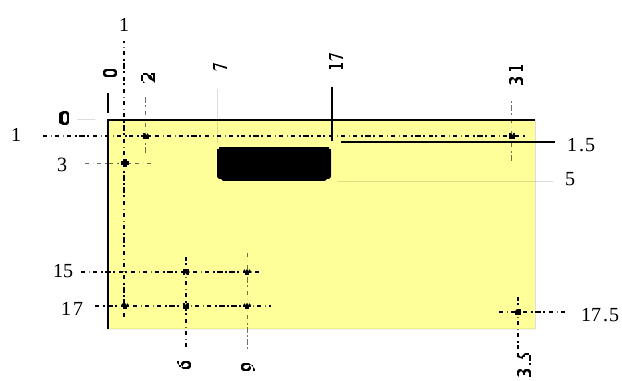


Workpiece drawing



How workpiece is held on the machine table

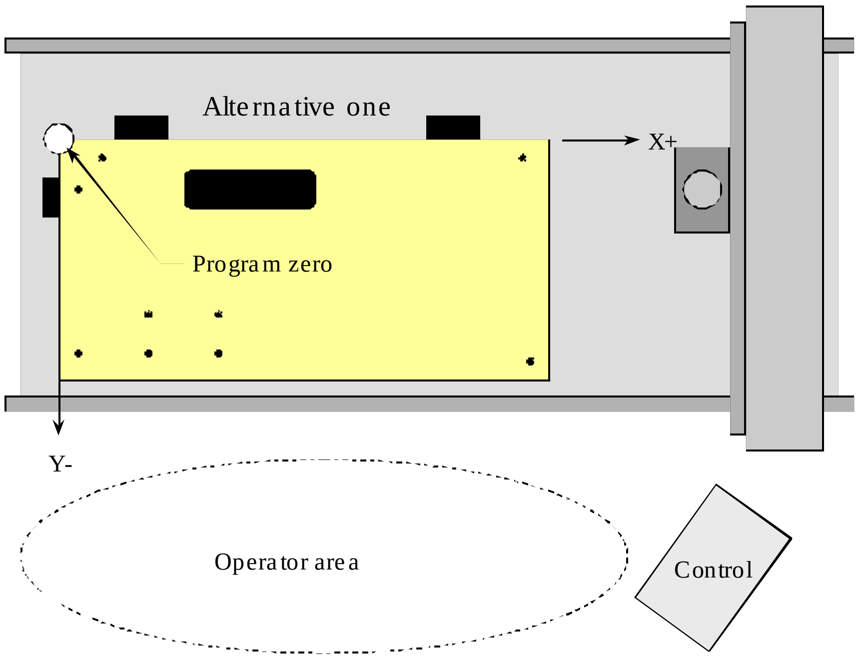
The datum surface dimensioning makes it easy to determine the XY program zero point (lower left hand corner of the workpiece). All X and Y coordinates will be positive (+). However, it just so happens that the orientation of this drawing nicely corresponds to the way the bump stops are located on the table. Consider the next example.



Workpiece drawing

Now the program zero point is the *upper* left hand corner of the workpiece. You can handle this problem in one of three ways. First of all, if all of the machining on this workpiece goes all the way through the workpiece (through holes and pockets), the workpiece can simply be flipped and run upside-down (top of the workpiece down). In this case, the workpiece can still be bump-stopped against the lower lefthand locators. But any machining does not go through the workpiece, another alternative must be found.

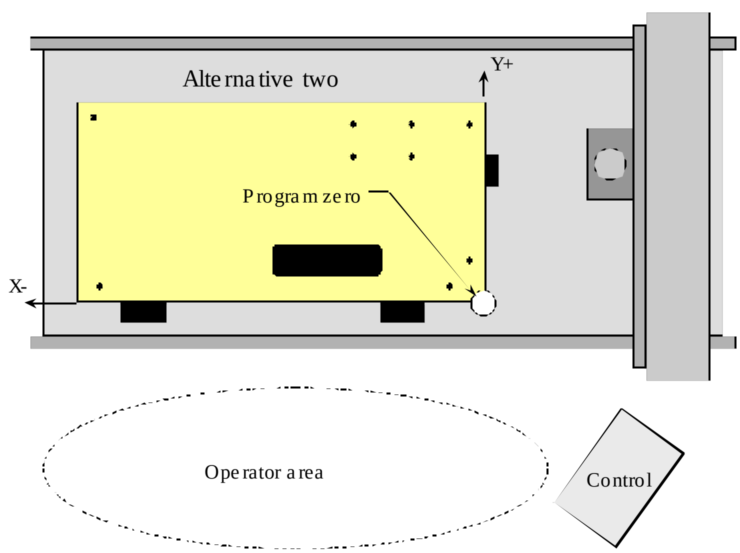
The second way to handle this problem is to move the locators. The next drawing shows one way to do so.



One way the workpiece is held on the machine table

Note that if this is done, the X locator can remain in the same position and the Y axis locators must be moved. Also, all Y axis coordinates will be minus (-).

A third alternative is to rotate the workpiece 180 degrees and locate it in the table from the right end. In this case only the (one) X axis locator must be moved for setup. If this alternative is chosen, you'll have to rotate the print in order to come up with the program coordinates. Also, all X coordinates will be minus (-).



Another way to locate the workpiece on the machine table

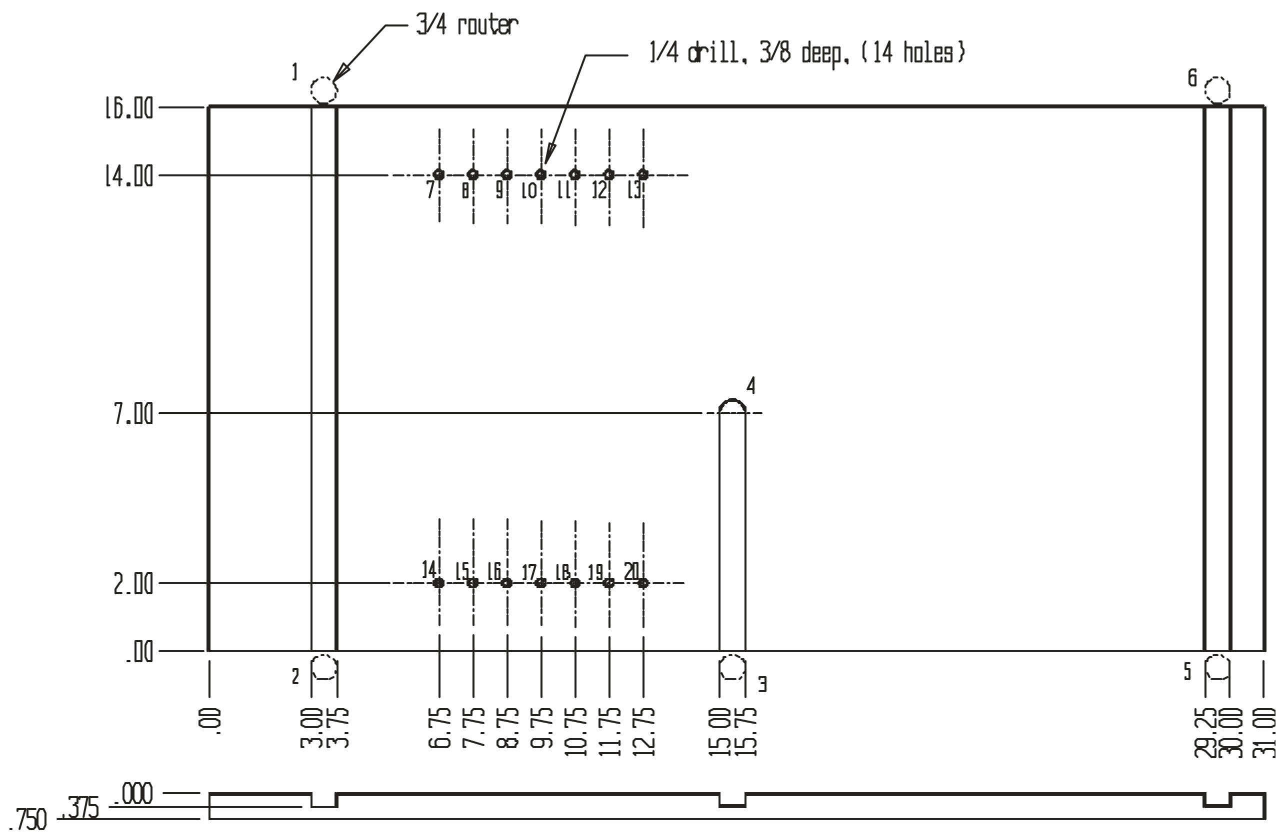
As you can see, selecting the appropriate program zero point commonly requires that you consider how the workpiece will be located on the machine table during machining. Only then can you come up with coordinates needed in the program.

1.5.4.1. What about Z?

As with XY, the programmer must designate a program zero position in the Z axis. We recommend that you make the program zero surface in Z the top surface of the workpiece. With this method, any machining that occurs in the workpiece (below the top surface) will require a minus (-) Z axis coordinate. We will use this technique for Z axis program zero selection throughout the course.

1.5.4.2. Example showing how coordinates are calculated

Here is a drawing dimensioned with datum surface dimensioning. Notice that program zero will be the lower left hand corner of the workpiece in X and Y. It will be the top surface of the workpiece in Z. This workpiece can be easily located on the machine table in the same relationship the drawing is made, meaning the X coordinates will be specified along the long side of this workpiece while the Y coordinates will be specified along the short side.



Notice that all points required in the program are numbered, including the positions needed for the 3/4 router. Here is a coordinate sheet showing the X, Y, and Z values needed in the program. Again, notice that all coordinates specified on the coordinate sheet are given relative to the program zero point. For points one through six (for the 3/4 inch router bit), notice that X coordinates are specified to the center of the dado.

#	X	Y	Z
1	3.375	16.475	-0.375
2	3.375	-0.475	-0.375
3	15.375	-0.475	-0.375
4	15.375	7.0	-0.375
5	29.625	-0.475	-0.375
6	29.625	16.475	-0.375
7	6.75	14.0	0.1/-0.45

#	X	Y	Z
9	8.75	14.0	0.1/-0.45
10	9.75	14.0	0.1/-0.45
11	10.75	14.0	0.1/-0.45
12	11.75	14.0	0.1/-0.45
13	12.75	14.0	0.1/-0.45
14	6.75	2.0	0.1/-0.45
15	7.75	2.0	0.1/-0.45

#	X	Y	Z
17	9.75	2.0	0.1/-0.45
18	10.75	2.0	0.1/-0.45
19	11.75	2.0	0.1/-0.45
20	12.75	2.0	0.1/-0.45

Coordinate sheet

1.5.5. Absolute vs incremental

When you are specifying coordinates from the program zero point (as we recommend), it is called the *absolute mode* of programming. The absolute mode is specified by a *G90* word in the program. Absolute mode differs from an older (yet still sometimes helpful) positioning mode of programming called *incremental mode*. With incremental mode, the programmer commands each movement from the tool's current position.

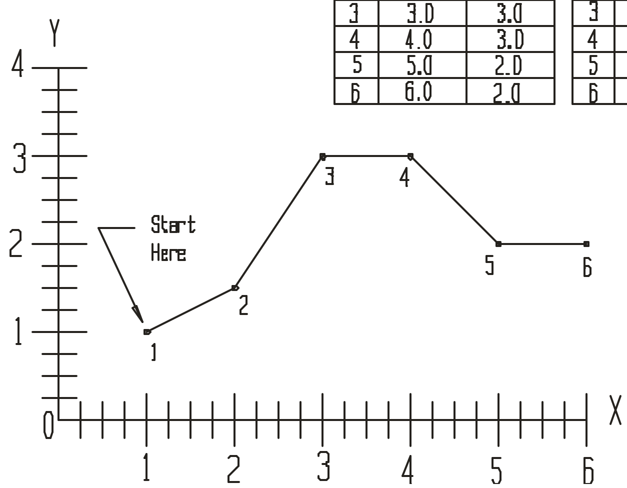
In the incremental mode (specified by *G91*) the programmer commands movements from the tool's *current position*. Each movement is specified as an incremental distance *and direction* from the tool's current location. While at first glance this may seem easier than working from program zero, you will find that incremental motions are very difficult to follow. Also, if the programmer makes a mistake in a series of incremental motions, *every* incremental movement from that point on will be incorrect. If the programmer makes the same mistake in a program written in absolute mode, only one movement will be incorrect. In the absolute mode, the machine will be back on track with the next correctly specified position. While there are some excellent applications for incremental mode (times when incremental mode can dramatically shorten a program's length), and we'll show them in key concept number six, *beginning programmers should concentrate on working exclusively in the absolute mode*. But keep in mind that any motion can be commanded in either the absolute or incremental mode. The next illustration shows this.

Absolute

#	X	Y
1	1.0	1.0
2	2.0	1.5
3	3.0	3.0
4	4.0	3.0
5	5.0	2.0
6	6.0	2.0

Incremental

#	X	Y
1	0	0
2	1.0	.5
3	1.0	1.5
4	1.0	0
5	1.0	-1.0
6	1.0	0



Movements made in both the absolute and incremental mode.

As you can see, absolute coordinates make sense. They match print dimensions. Incremental motions do not. They are nothing more than a whole series of small movements, each taken from the tool's previous position.

1.5.5.1. Warning about thinking incrementally

We must warn you that beginners tend to make mistakes with regard to absolute and incremental modes. Beginners tend to *think incrementally*, specifying motions as distances from the tool's last position when they should be specifying positions from the program zero point. When coming up with coordinates needed for your program ask yourself, *"to what position will the tool be moving in this command?"* This position must be specified relative to the program zero point. Beginners tend to ask themselves the wrong question. They tend to ask, *"how far should the tool move in this command?"* That's thinking incrementally.

1.5.6. Be sure to use the decimal point when specifying axis positions

Though decimal point programming is introduced in lesson three, it is a very important feature for coordinate entry and we want to describe its use in further detail. As stated, whenever you want to make the machine move along one or more of the axes, you will be specifying the *coordinate* to which you want to make the tool move. To do this, you must always specify the letter address of the axes you wish to move (X, Y, and/or Z) along with the coordinate position to which the tool must move. The coordinate position value tells the control where along the axis to stop the tool.

Current controls allow you to specify this coordinate position with a decimal point. Older controls (over about 20 years old) do not allow the decimal point. You must remember to use a decimal point in *all* of your program's coordinate positions. One technique you can use to help you remember the decimal point is to include at least one digit to the right and left of the decimal point. When writing whole numbers, for example, carry your value to the first zero to the right of the decimal point. Write

X3.0

for example, instead of

X3.

In similar fashion, include the zero to the left of the decimal point when writing values under one. Write

X0.5

instead of

X.5

Again, this will help you keep from forgetting to write or type decimal points when writing or entering programs.

As you have seen, it is much easier to read a program if it contains coordinates with decimal points. Here are a few examples of CNC words including decimal points.

3.125 in the Y axis would be specified as Y3.125

2.12731 in the X axis would be rounded to X2.1273

3 inches in the Z axis would be specified as Z3.0

4.75 in the X axis would be specified as X4.75

Though we have been discussing coordinate positions, you will also find that as decimal point can be included in other types of CNC words when it is feasible to do so. For example, the feedrate command (F word) also allows a decimal point.

As mentioned, older controls (over about twenty years old) do not allow decimal point programming. Fortunately, CNC is relatively new to the woodworking industry, meaning that relatively few older CNC routers exist that do not allow decimal point programming.

With these older machines that do not allow decimal point programming, a fixed format must be used with real numbers (numbers including a decimal portion). Keep in mind that this method of designating positions is still allowed, even on current model controls. Here are some examples that assume *four place trailing zero fixed format*.

3.125 in the Y axis would be specified as Y31250

2.12731 in the X axis would be rounded to X21273

3 inches in the Z axis would be specified as Z30000

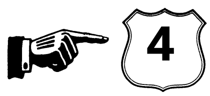
4.75 in the X axis would be specified as X47500

Note that in all cases, the value is carried out to the fourth place to the right of the decimal point. The control will automatically place the decimal point four places to the left of the right-most digit.

This should give you a clue about what will happen if you *forget* to include a decimal point in a word that requires it. If for example, you intend an X position of three inches (should be specified X3.0) and forget the decimal point (specifying X3), the control will automatically place the decimal point four places to the left of the right-most digit (3 in our case). It will take the value specified in X3 as X0.0003 instead of X3.0. Again, it is very important that you include the decimal point in words that require it.

While most manual programmers prefer to use decimal point programming for obvious reasons, note that some computer aided manufacturing (CAM) systems still generate CNC programs without decimal points. If your company owns this kind of CAM system, you must, of course, be able to recognize the value of each CNC word, even if it does not have a decimal point.

STOP!! Do practice exercise number:



Lesson 5

1.6. Locating the program zero point for a program

At this early point in the course, it is much more important that you understand the program-zero-related topics presented thus far than it is to know how program zero is located and assigned. However, you must know that just because you want the program zero point to be in a certain place doesn't mean the CNC control will automatically know its chosen location. In reality, a conscious effort must be made to *assign* the program zero point. Think of it this way. The programmer determines a logical position for the program zero point. All coordinates going into the program are specified from this point. However, the CNC machining center or router *control* must also know the location of the program zero point in order to make the machine move correctly. In essence, when the setup person *assigns* program zero, they are *marrying the program to the workholding setup*.

By one means or another, the control must be told the location of the program zero point in order to make the machine move properly. In lesson 6, you will learn that there are two ways by which program zero can be assigned. With either method, the techniques shown in this lesson can be used to *find* or *measure* the program zero point at the machine.

Both methods we show in lesson 6 require that you know the distance between your selected program zero point and the machine's spindle while all axes of the machine are resting at the machine's *zero return position*. The method we show to locate the program zero point for the machine is rather crude. It requires measurements to be taken right on the machining center after the workholding setup is made. While there are techniques that can minimize (or even eliminate) these tedious, time consuming, and error prone measurements, the method we show does make it easy to understand what the program zero assignment values represent.

1.6.0.1. <u>What is the zero return position?</u>

There are two *zero positions* that will be of constant concern to you while you are working with Fanuc-controlled CNC routers and machining centers. First is the previously discussed *program zero position*. You now know that program zero is the location on your workpiece from which all coordinates going into your program are taken.

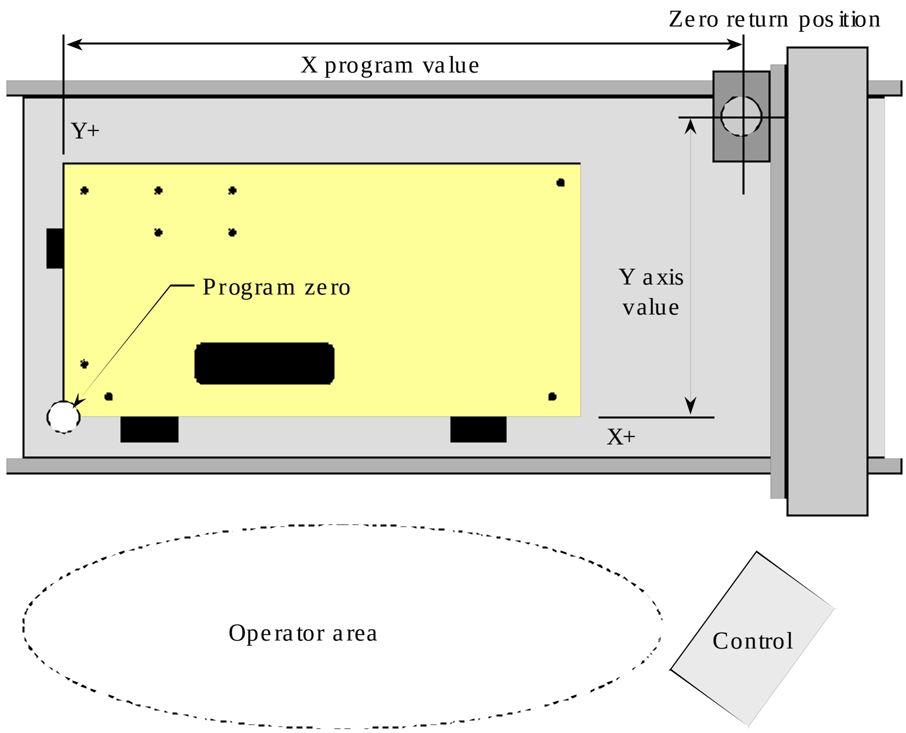
The second zero point is the machine's *zero return position* (some people call this position *home position*, *machine zero*, *grid zero position*, or *reference position*). The zero return position is a very accurate location along each of the machine's axes which acts as a point of reference for program zero assignment. In fact, the machine start-up procedure for most machines requires that you send each axis to its zero return position. When the machine is sent to its zero return position, three *axis origin* lights come on (one for each axis) to indicate that the machine is resting at its zero return position.

For most CNC routers and woodworking machining centers, the zero return position is close to the *extreme plus limit* of each axis. When a CNC router is resting at zero return position, the column is all the way to the right in X (as viewed from the operator station), the headstock is as far as it can move away from the operator in Y, and the quill is all the way up in Z.

The zero return position is the location from which you will measure the program zero point. Furthermore, depending upon how program zero is assigned, the zero return position may also be the location from which you will actually execute your program (more on this in lesson 6).

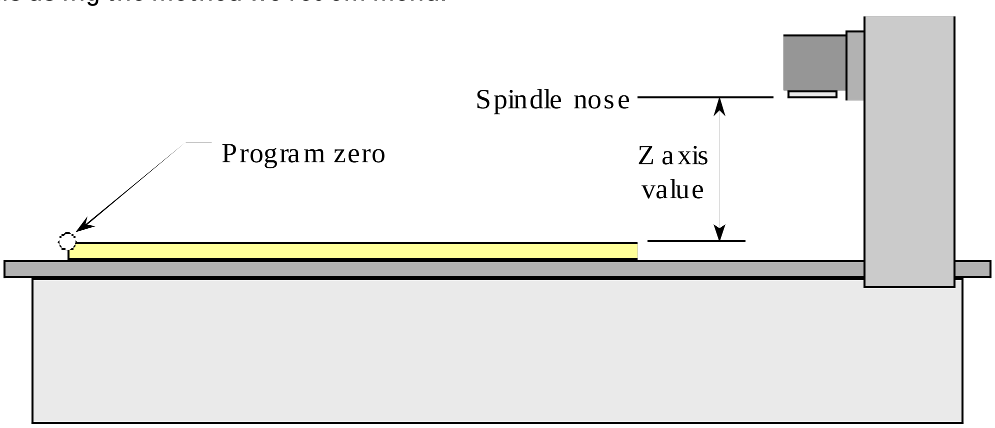
1.6.1. How to measure the program zero point location on the machine

The next illustration shows the X and Y program zero assignment values that must be determined before program zero can be assigned. Notice that the machine is resting at its zero return position in this illustration.



XY axes program zero assignment values

Again, the X and Y axis program zero assignment values are the distances between the program zero point and the center line of the spindle while the machine is resting at its zero return position. The Z axis program zero assignment value is dependent upon how you intend to use a feature called *tool length compensation*. We discuss this tool length compensation in key concept number four. By our recommended method, the program zero assignment value in the Z axis will be the distance between the program zero point and the spindle nose in the Z axis. The next illustration shows the program zero assignment value for the Z axis using the method we recommend.



Z axis program zero assignment value

There are *many* ways to actually measure the program zero assignment values at the machine during setup. But remember that these measurements take time, and if you can come up with a way to eliminate these measurements, you can save a lot of setup time. We'll discuss getting more efficient a little later. For now, we just want to show *one way* these values can be measured.

Each method requires that you use the machine's *position display* of the display screen. This page is much like a *digital readout* for any kind of manual equipment. Each method also requires that you manually move each axis in a precise manner. Most machines have some kind of *handwheel* to allow precise movements.

1.6.1.1. Measuring the X and Y axis program zero assignment values:

Some kind of alignment tool must be placed into the spindle to help with these measurements. While a variety of tools can be used for this purpose, say you place a metal dowel pin of known diameter into a Jacob's chuck in the spindle. Say the pin is 0.25 inch in diameter (having a radius of 0.125). The dowel pin will be used as a kind of aligning bar. Whenever you have the dowel pin touching a surface, you'll know that the spindle center is precisely 0.125 from the surface. While there are much more accurate devices available to help with this kind of measurement (edge finders and dial indicators, for example), a dowel pin will usually be sufficient giving the accuracy required of most woodworking setups.

1) Place a 0.25 diameter dowel pin in the spindle

2) Manually move the dowel pin (in X, Y and Z) close to the surface to touch in X

3) Using the machine's handwheel, cautiously position the dowel pin flush with the X axis program zero surface (use some kind of feeler to sense contact).

4) Make the X axis display read zero

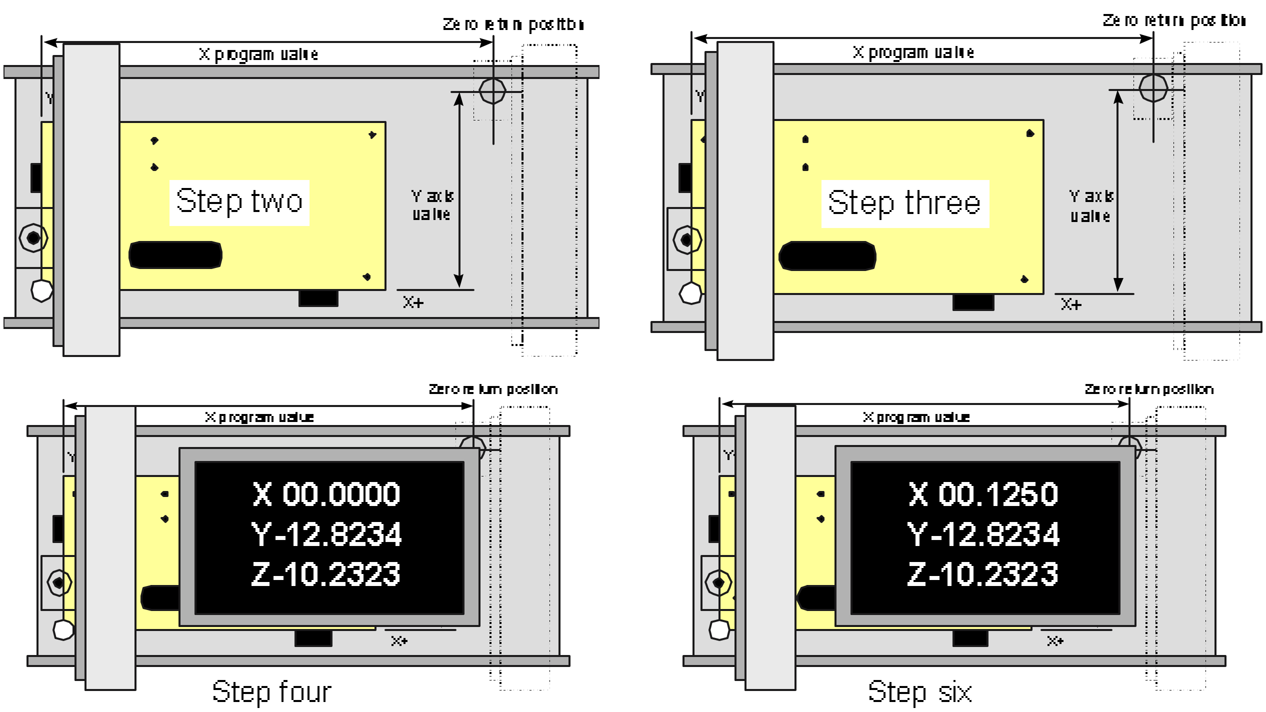
5) Move the dowel pin up above the surface in Z

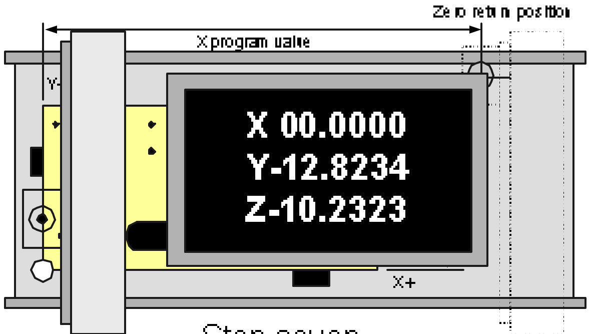
6) Move in X toward the surface by the dowel pin's radius (0.125 for a 0.25 dowel pin)

7) The center of the spindle is right on the program zero point in X. Now make the X axis display read zero again.

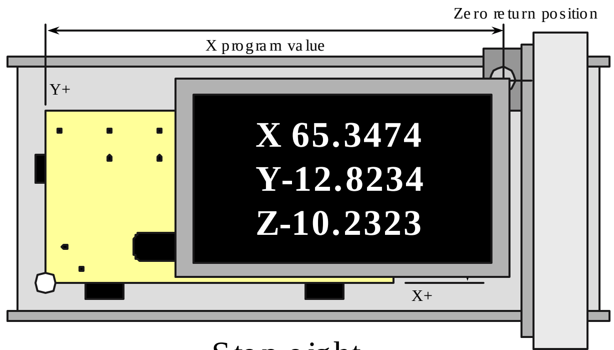
8) Send the machine to its zero return position in X (the X position display will follow along). When the machine gets to zero return position, the X axis display will be showing the distance from program zero to the center of the spindle at zero return position in X.

9) Repeat steps 2 through 8 for the Y axis





Step seven



Step eight

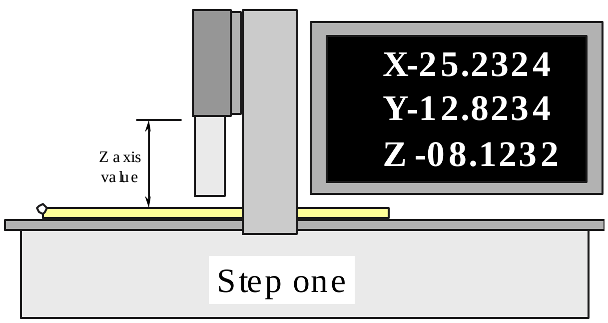
1.6.1.2. <u>To measure the program zero assignment value in Z:</u>

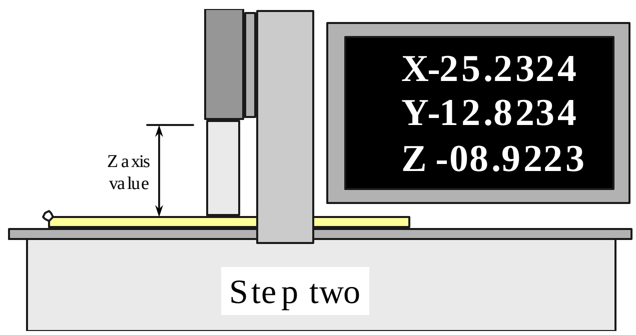
1) With nothing in the spindle, manually move the nose of the spindle down close to the program zero surface in Z

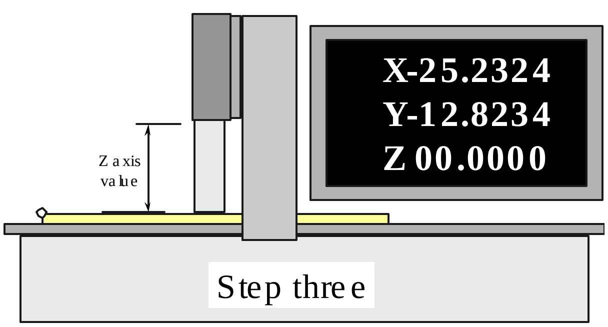
2) Cautiously touch the nose of the spindle to the program zero surface

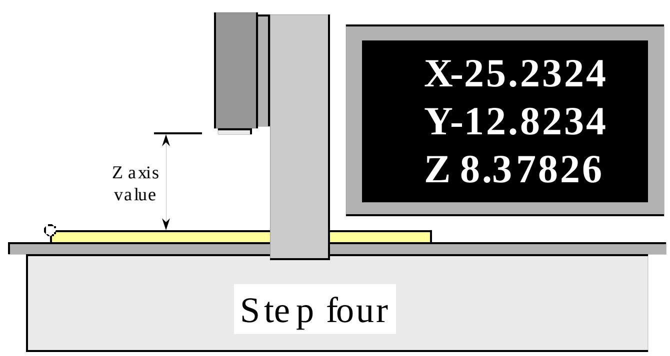
3) Make the Z axis display read zero

4) Manually send the machine to its zero return position. When the Z axis reaches the zero return position, the Z axis display will be showing the distance from program zero to the nose of the spindle in Z.









Note: Again, the procedure for measuring the Z axis distance may change based on your method of using tool length compensation. More on this during key concept number four.

1.6.1.3. Do you have to physically *measure* the program zero point?

While the steps just given should help you understand how program zero assignment values can be measured, measuring program zero for every setup will take a lot of setup time. If you can eliminate these measurements, you can dramatically reduce setup time. Due to the consistent nature of workpieces machined on CNC routers and machining centers (square or rectangular flat stock), many companies can mount location blocks (stops) in a semi-permanent manner. If the stop-blocks don't move from one setup to another, the program zero assignment values in X and Y won't change.

Additionally, once the Z axis program zero assignment value has been determined for one stock thickness, it will remain the same every time a workpiece of that thickness is run. And of course, it will be quite easy to determine the Z axis program zero assignment values for workpieces of other thicknesses with simple addition and subtraction (no need for more Z measurments). In reality, most CNC router-using-companies don't have to measure program zero for every setup. By applying a little ingenuity, some *never* have to measure it.

Also keep in mind that most controls allow you to *program* the program zero assignment values, meaning you can even eliminate the *entry* of program zero assignment values. More on ways to improve setup efficiency will be presented in key concept number four.

STOP!! Do practice exercise number:

5

Lesson 6

1.7. The two ways to assign program zero

In this lesson we show the techniques used to actually assign the program zero point. As stated in lesson five, it is *much* more important at this time that you understand how to determine the program zero point's position and work in the absolute mode than it is to actually assign program zero.

Admittedly measuring and assigning program zero is more a setup person's responsibility than a programmer's (and we do show these techniques from a setup person's point of view in key concept number seven). However, in most companies the CNC programmers is expected to provide direction to the setup people and operators. Indeed, in key concept number two, you'll see that programmers are commonly expected to provide setup and production run documentation. In order to do so, they must know as much as setup people about program zero assignment.

By far, the best way of assigning program zero is to do so with a feature called *fixture offsets*. In fact, the only reason *not* to use fixture offsets is if your machine does not have the feature. If your machine has fixture offsets, we urge you to use fixture offsets to assign program zero. Unfortunately, not all controls have fixture offsets. Without fixture offsets, the only way to assign program zero is to do so within the program with a *G92* command. Before going much further, you should confirm whether or not the machines you'll be working with have fixture offsets. (Ask an experienced person in your company or school or check with your machine tool builder.)

With either method of assigning program zero, *you use the program zero assignment values discussed in the last lesson* for program zero assignment.

1.7.1. Advantages of assigning program zero with fixture offsets

As stated, fixture offsets make a much better way of assigning program zero. Here are some reasons why.

1.7.1.1. Safety

A G92 command within the program simply tells the control how far it is *from* program zero *to* the spindle *at the current time*. When this command is used to assign program zero, each axis of the machine *must* be in a previously planned location *before* the cycle can be activated. If the machine is out of position for any reason, the machine will *not* move to the correct location. If the machine is moved in a minus direction (in any axis) from its planned starting point, the distance the machine will traverse will be greater than it should be, in effect, causing a collision between the tool and workpiece. *This is the single largest cause of crashes on machining centers: When program zero is assigned in the program and the machine is out of position when the cycle is activated!* With fixture offsets, the machine need not be in a previously planned position prior to activating the cycle. In effect, the control will track the machine's current position and take it into consideration when the program zero assignment is invoked.

1.7.1.2. Ease of use

If program zero is assigned in the program, the operator must be certain that the machine is in the proper location prior to activating a cycle. This could require tedious manual movements to the machine's zero return position. With fixture offsets, the operator need not worry about sending the machine to a specific location (other than possibly the machine's tool change position) prior to activating the cycle.

1.7.1.3. Efficiency

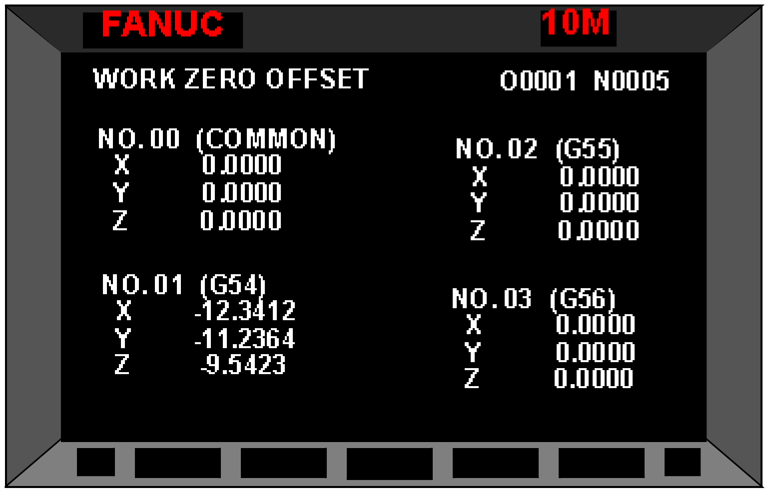
Since the machine must be in a precise location prior to activating a cycle if program zero is assigned in the program, many wasted motions must be programmed the beginning of the program, possibly during each tool change, and at the end of the program. These motions just to get the machine to its proper starting position, and can be very time consuming.

1.7.1.4. Rerunning tools

The operator will often need to re-run a given tool in the program. Doing so if program zero is assigned in the program can be somewhat difficult, since the machine must be properly position prior to rerunning the tool. Additionally, the G92 command must be executed prior to re-starting. When assigning program zero with fixture offsets, no prior positioning movements are necessary.

1.7.2. Assigning program zero with fixture offsets (G54-G59)

Fixture offsets allow you to separate the program zero assignment from the CNC program. The program zero assignment values (shown in lesson five) will be entered on a special display screen page. This drawing shows the fixture offset page.



Fixture offset page of a popular CNC router control

The techniques to actually enter program zero assignment values into fixture offsets will be shown in key concept numbers four and nine. Fixture offsets are specified *from the machine's zero return position to the program zero point.* Since the zero return position is close to the plus limit of each axis on most CNC routers, the polarity of fixture offset values will almost always be minus.

Another benefit of fixture offsets is that you can have up to (at least) six coordinate systems. This means up to six program zero points, or coordinate, systems, can be assigned. This can be used for multiple workpieces in a setup. For now we will limit our discussions to utilizing this feature for only one coordinate system.

Activating fixture offsets from the program is very simple. *G54* specifies that you want to work in coordinate system number one. *G55* is for coordinate system number two, and so on through coordinate system number six (*G59*). When the control reads a G54 command, it looks to the values included in the G54 fixture offset to attain the distances from zero return position to program zero.

Using fixture offsets (G54-G59) to assign program zero is must safer than using G92. Since all measurements are taken to the zero return position, and since the measured values are *not* part of the program, the control will *always* keep track of the machine's position. This means that if the machine is *not* at the planned starting location when the cycle is activated, the control will *still* send the tool to the proper location. As we said, if this were done while the G92 technique was used, a crash could occur.

For now, that's enough about fixture offsets. We'll discuss their special use for assigning multiple program zero points in key concept number four.

1.7.3. Assigning program zero in the program

As stated, this is the less desirable of the two methods used to assign program zero. On some controls, it may be the *only* way to assign program zero.

Additionally, some programmers assign program zero within the program with G92 to maintain compatibility with older equipment. Some companies have several CNC routers that range in age from very old to quite new. A company's older machines might require the G92 method of assigning program zero. To maintain compatibility throughout the shop, a programmer may decide to stick with its use. However, we urge people *not* to use G92 command unless you don't have fixture offets.

When G92 is used to assign program zero within the program, the programmer will simply include the measured distances *from program zero to the machine's starting position* (shown in lesson 5) within the G92 command. This command is given very close to the beginning of the program.

Notice that the distance is taken *from* the program zero point *to* the machine's starting position. For almost all CNC routers and machining centers, this means the polarity of all G92 values will be positive.

Here is an example of a G92 command assuming the program zero assignment values are 10.0712 in X, 13.7192 in Y, and 12.9281 in Z.

```
N005 G92 X10.0712 Y13.7192 Z12.9281
```

Again, notice that the X, Y, and Z values in the G92 command are the distances measured *from* program zero *to* the center line of the spindle in X and Y and to the nose of the spindle in Z (shown earlier) and will almost always be positive values.

When the control reads this command, it sets the *absolute position displays* on the control screen to the X, Y, and Z values of the G92 command itself. Basically, the control is being told that the distance *from* program zero *to* the *current* center line of the spindle in X is 10.0712 inches. In Y the current distance *from* program zero *to* the center line of the spindle is 13.7192. And in Z, the distance *from* program zero *to* the current position of the spindle nose is 12.9281 inches.

Notice the word *current* in each of the three previous sentences. This is a very important word. The use of the G92 command in the program requires that the operator be extremely careful with the program's starting point. If the program is intended to be run with the machine resting at the zero return position, the operator *must* be sure the machine is at zero return position *before* they press the cycle start button. If the operator moves the machine axes out of position (maybe to check a tool in the spindle), when cycle start is pressed the control would assume that the G92 command dimensions are the distance from the program zero point to the machine's current position. If the machine was moved toward the workpiece, the control will try to send the tool through the workpiece, causing a collision. Again, this is the biggest source of crashes; when the operator presses the cycle start button while the machine is not at its proper starting point.

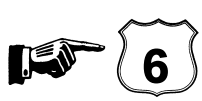
When assigning program zero in the program, the machine's zero return position does make an excellent position from which to start, especially for beginners. Though it may not be the most efficient position from which to start the program, at least the operator can easily confirm that the machine is at its proper starting position before the cycle is activated. Three green axis origin lights will come on when the machine is at zero return position, indicating to the operator that it is safe to start the cycle. If zero return is used as the program's starting point, it is wise to make a command to send the machine to the zero return position *prior* to the G92 command. This will help save a crash if the machine has been moved out of position.

One command that can be used to send the machine to its zero return position is as follows (more on how this command works during key concept number five):

```
N005 G91 G28 X0 Y0 Z0
N010 G92 X10.0712 Y13.7192 Z12.9281
```

Notice that this command is given just prior to the G92 command. The programmer can rest assured that the machine will be at the zero return position when the G92 is read. It is a very nice safety enhancing technique.

STOP!! Do practice exercise number:



Lesson 7

1.8. Introduction to programming words

Now is as good a time as any to introduce the various words involved with programming. If you are a beginner looking at these words for the first time, you may want to read this section several times to get better acquainted with these word types. Note that we are *not* asking you to memorize these CNC words. When we get into the program formatting concept (concept number five), we will be giving you a way to simply look at each word and try to remember its function. In key concept number five, you will not have to be able to come up with all the words used in programming completely on your own.

Also, this section of the course is simply intended to *introduce* each word, not to give you an in-depth description. You will find that certain words are seldom used, meaning you will have little or no need for them. Other words are constantly used, and you will soon have them memorized after writing a few programs.

Some programming words have more than one function, depending on command format. We will be showing you the *primary* function of the word next to the “A” description and any *secondary* use for the word next to the “B” description.

As a beginner, do not let the number of different words intimidate you. You will find that most of them are aptly named, having the letter stand for something that is easy to remember, like “S” for spindle speed, “F” for feedrate, and “T” for tool station. It should not be too difficult to remember the function of most words. Additionally, you will find that only thirty to forty words are used consistently when programming, so *look at learning CNC programming as like learning a foreign language that has only thirty or forty words.*

1.8.1. Note About Decimal Points

Remember that all current controls allow the programmer to include a decimal point for those word types that are feasible to use a decimal point with. Older controls do not. Those word types that *do* allow decimal points are:

A, B, C, X, Y, Z, I, J, K, F, Q, R, A, B, and C

O This is the word most controls use for a program number. All controls discussed in this course (Fanuc and Fanuc-compatible) allow the user to store multiple programs in the memory of the control. The programmer will be assigning the program a number from 0001 through 9999. The O word will be the very first word in the program. No decimal point is allowed with the O word.

N This word specifies a sequence number. A sequence number is used for program line identification. It allows the programmer to organize each line in his program by number. This allows easy editing of a program at the machine. Sequence numbers are not required to be in any particular order and can even repeat in the program. Also, they do not even have to be in the program at all. But for the sake of organization purposes, we recommend that beginners include them in the program and place them in an understandable order. For examples in this text, we will go by fives to allow for extra lines to be inserted if needed. No decimal point is allowed with the N word.

G This code specifies what is called a preparatory function. Preparatory functions allow various modes to be set in the program. There are many G words, but only a few that are used on a consistent basis. Most are modal, but a few are one-shot (non-modal). For a list of all G codes, see the list at the end of this handout. No decimal point is allowed with most G words.

X A. The primary use for the X word is to designate a coordinate along the X axis. The X word can be specified with a decimal point or without. With the decimal point, an X position of 10 inches would be specified “X10.0". Without the decimal point the X word will go back to an old method of programming called fixed format. Without the decimal point an X position of 10 inches would be specified ”X100000", and the decimal point would be *assumed* to be four places from the right digit. For beginning programmers, it is much easier to program with the decimal point, but you must be sure to include it in every X word, or the machine will revert to the fixed format. The same thing goes for the Y, Z, R, I, J, K, Q, and F words.

B. The secondary use for the X word is that it can be used to specify the length of time in a dwell command (see G04).

Y The Y word specifies a position along the Y axis. All decimal point related functions are the same as for the X word.

Z A. The primary use for the Z word is to specify a position along the Z axis. All decimal point related functions are the same as for the X word.

B. The secondary use for the Z word is to specify the hole bottom position in a canned cycle command. Canned cycles are discussed in key concept number six.

A If your machine has a rotary table of some kind (*not* a simple indexing device), the rotation of the device is considered a true axis of motion, just like X, Y, and Z. Depending on the style of the machine (horizontal vs vertical) the rotary axis can be designated with the letter address A, B, or C. Generally speaking, vertical machining centers that have the rotary axis call the rotary axis the C axis. Horizontal machining centers call the rotary axis the B axis. However, some builders do not follow this standard. If your machine has a rotary table, you must consult your builder’s manual to find out what they have named the rotary axis.

B See the description for “A”.

C See the description for "A".

R A. The primary use for the R word is to specify the radius of a circular move. All decimalpoint related functions are the same as for the X word.

B. The secondary use for the R word is to specify the *rapid plane* for a canned cycle command. Canned cycles are discussed in key concept number six.

I A. I, J, and K words are the old way to specify the arc in a circular move. While they are still effective, we strongly recommend that the beginner concentrate on using the R word to specify the arc in a circular move. (It is much easier!) If using I, J, and K, these words specify the distance and direction in X, Y, and Z (I=X, J=Y, K=Z) from the starting point of the arc to the center of the arc. While I, J, and K do allow a full circle to be generated in one CNC command, they are quite difficult for a beginner to understand. The R word is much more understandable. The I, J, and K follow the same decimal point rules as the X word.

J See description of "I".

K See description of "I".

Q The Q word is used with peck drilling canned cycles to specify the peck depth for each pass. Canned cycles are discussed in key concept number six. All decimal point related format related to the "X" word apply to the Q word.

P The P word is used to specify the length of time in seconds for a dwell command. Dwell commands are used to make the axis motion (for all axes) pause for a specified length of time. The P word specifies this length of time. A time of three seconds would be specified as P3000 (with NO decimal point). Note that the fixed format of the P word has *three places* to the right of where the decimal point would be. Other examples: P2500 = 2.5 seconds, P500 = .5 second, and P10000 = 10 seconds. No decimal point is allowed with the P word.

B. The secondary use for the P word is with subroutines to specify the subroutine program number to be searched. Subroutines are discussed in key concept number six. No decimal point is allowed with the P word.

L A. The L word is used with sub-programming techniques (discussed in key concept number six) to specify the number of times a sub-program should be executed. No decimal point is allowed with the L word.

B. The L word can also be used with canned cycles (discussed during key concept number six) to specify the number of holes to machine. No decimal point is allowed with the L word.

F The F word specifies the desired feedrate in inches per minute (IPM) or millimeters per minute. This word is used to tell the control the desired traverse rate in a cutting command (G01, G02, and G03). The F word allows a decimal point, so a feedrate of 3-1/2 inches per minute would be programmed "F3.5". However without the decimal point, the fixed format is slightly different than the X word. "F350" would be the "fixed format" for a feedrate of 3.5 inches per minute. Note that the meaning of the "F" word changes based on input mode. In the inch mode (G20), the feedrate is specified in inches per minute. In the metric mode (G21), the feedrate is specified in millimeters per minute.

S

The S word specifies a spindle speed. Most current machines allow the programmer to specify the exact RPM they desire. That is, a spindle speed of 350 RPM would be programmed “S350". No decimal point is allowed with the S word. Though it may not be extremely important at this early point in the course, if the machining center has more than one spindle range, many machines will have the control key on the S word value to determine the appropriate spindle range. Say for example, your machine has two spindle ranges. The low range runs from 30-4,000 rpm. The high range runs from 4,001 through 10,000 rpm. For this machine, a speed selection of S300 would automatically force the machine to select the low range. A spindle speed selection of S5000 would have the machine select the high spindle range.

T

With most machines, the T word specifies a two digit tool station to be placed in the *load position* of the tool changer magazine. While one tool is cutting, it is common to select the T word of the next tool so that when the tool in the spindle is finished machining, the next tool will be ready. No decimal point is allowed with the T word. Note that machine tools vary with regard to how the T word is specified. These differences are discussed in key concept number five.

M

The M word specifies a series of *miscellaneous functions*. You can think of M words as programmable *on/off switches* that control functions like coolant and spindle rotation. For a list of all M words, see the list at the end of this lesson. Note that machine tool manufacturers will select their own set of M words. While there are many standard M word numbers, you must consult your own machine tool manufacturer’s manual/s to find the exact list for your particular machine/s. No decimal point is allowed with the M word.

D

The D word specifies the offset number to be used with router radius compensation. Router radius compensation is discussed in key concept number four. No decimal point is allowed with the D word.

H

The H word specifies the offset number to be used with tool length compensation. Tool length compensation is discussed in key concept number four. No decimal point is allowed with the H word. Generally speaking, the programmer should make the H word the same number as the tool station number.

EOB

This is a command terminator. It tells the control that the command ends. If entering programs through some form of text editor, this character is usually entered by a carriage return. If entering programs through the control’s keyboard, it is entered by the key labeled EOB. This character shows up on the display screen as a semi-colon or an asterisk.

/

This is called the *optional block skip* word (also called *block delete*). It works in conjunction with a on/off switch on the control panel labeled optional block skip or block delete. If the switch is on when the control reads the slash code, the control will ignore the command that begins with the slash code. If the switch is off, the control will execute the command in the normal manner. More on this function during key concept number six.

1.9. G and M words

Here we continue the description of the various words used in programming. We will state again that it is *not* necessary to try to memorize all of these words. This section of the manual can be used as reference to help you easily determine the meaning of each G and M code.

1.9.1. G words

As mentioned in the previous discussion, G words specify what are called preparatory functions. They prepare the machine for what is to come. They set modes. Note that only *three compatible G words are allowed per command*.

1.9.1.1. Note on Option G Codes

Note that many of the G words are listed as options (specified in the list). It is impossible to tell whether a given option G code is included in your control or not by just looking at this list. Most machine tool builders include a standard package of options when they purchase the control from the control manufacturer Our list show what one popular *control manufacturer* calls options (Fanuc). Probably when your company purchased the machine from your machine tool builder *many* more G codes came with the machine. If there is any question as to whether your machine has any one particular G code option, you can perform a simple test at the machine to find out if the G code is available to you (or you can call your builder to find out if the G code was included).

To make the test for any option G code, simply command the G code in the MDI mode (techniques given in the operation handbook in key concept number nine). You need not even specify the correct format for the G word. If you receive the alarm UNUSABLE G CODE or G CODE NOT AVAILABLE, your machine does not have the G code. If you receive no alarm or if the alarm is related to the format of the G code, the G code should be available for you to use.

1.9.2. A reminder about initialized G codes

We also state whether the G word is initialized, meaning whether they are in effect when you turn the power on to the machine.

Again, beginners should not be intimidated by the number of G words. You will find that many are seldom used, and to just know they are available is more than good enough for now.

G word	Description	Status	Initialized
G00	Rapid motion	Std	Yes
G01	Straight line cutting motion	Std	No
G02	Circular cutting motion (CW)	Std	No
G03	Circular cutting motion (CCW)	Std	No
G04	Dwell Command	Std	No
G09	Exact stop check (one shot)	Std	No
G10	Offset input by programmed command	Option	No
G17	XY plane selection	Std	Yes
G18	XZ plane selection	Std	No
G19	YZ plane selection	Std	No
G20	Inch mode	Std	Yes
G21	Metric mode	Std	No
G22	Stored stroke setting (safety zone)	Option	No
G23	Stored stroke cancel	Option	Yes
G27	Zero return check	Std	No
G28	Zero return command	Std	No
G29	Return from zero return position	Std	No
G30	Second reference point return	Option	No
G31	Skip cutting (for use with probes)	Option	No
G40	Router radius compensation cancel	Std	Yes
G41	Router radius compensation left	Std	No
G42	Router radius compensation right	Std	No
G43	Tool length compensation	Std	No
G44	Tool length compensation for minus input	Std	No
G49	Tool length compensation cancel	Std	Yes
G52	Return to base coordinate system	Option	No
G53	Temporary shift to machine coord. system	Option	No
G54	Fixture offset select for system #1	Option	No
G55	Fixture offset select for system #2	Option	No
G56	Fixture offset select for system #3	Option	No
G57	Fixture offset select for system #4	Option	No
G58	Fixture offset select for system #5	Option	No
G59	Fixture offset select for system #6	Option	No
G65	Custom macro call	Option	No
G66	Custom macro modal call	Option	No
G67	Cancel custom macro modal call	Option	No
G68	Coordinate system rotation	Option	No
G69	Coordinate system rotation cancel	Option	Yes
G73	Peck drill cycle for steel	Std	No
G80	Cancel canned cycle	Std	Yes
G81	Standard drilling cycle	Std	No

G word	Description	Status	Initialized
G82	Counter boring cycle	Std	No
G83	Peck drilling cycle to clear chips	Std	No
G90	Absolute programming mode	Std	No
G91	Incremental programming mode	Std	Yes
G92	Program zero designator	Std	No
G98	Return to initial plane (G73-G89)	Std	Yes
G99	Return to “R” plane (G73-G89)	Std	No

1.10. Typical M codes for a CNC router or machining center

Note that these are only the *typical* M words for CNC routers and machining centers. This is not a complete list. Your machine tool builder will surely have additional M codes listed in their programming/operation manuals. You must check these manuals for the complete list. Note that only one M word is allowed per command on most controls.

M CODE	DESCRIPTION
M00	Program stop
M01	Optional stop
M02	End of program (does not rewind memory))
M03	Spindle on in a clockwise direction
M04	Spindle on in a counter clockwise direction
M05	Spindle stop
M06	Tool change command
M30	End of program (rewinds memory)
M98	Sub program call
M99	End of sub program

Other M Codes for your machine (found in your machine tool builder’s manuals)

____ ________________________________

____ ________________________________

____ ________________________________

____ ________________________________

____ ________________________________

____ ________________________________

____ ________________________________

____ ________________________________

1.10.1. Why Learn Manual Programming?

As you are beginning to see, the conventional CNC control requires a rather cryptic form of CNC program. Without previous exposure to CNC programming, the beginner will find even the simplest CNC program rather difficult to understand. For this reason, there are those in the industry who feel that manual programming is a thing of the past. They contend that there are alternative to manual programming that eliminate the need to learn it.

Admittedly, there are alternatives to manual programming. Though this is the case, the beginning programmer must grasp a basic understanding of manual programming in order to continue on to the alternatives. There are three important points we want to make related to the alternatives to manual programming.

First, there is a form of CNC control called a *conversational* control. Conversational CNC router and machining center controls include the Fanuc 0MF, 11MF, 15MF, and 16MF. This newer form of CNC control allows the operator to input programs right at the machine in a more understandable and *graphic* manner than with manual programming. However, there is quite a controversy in the industry dealing with whether conversational controls should be used. Many companies, for example, do not wish their CNC operators to be creating CNC programs. This, coupled with the fact that conventional CNC controls requiring manual programming currently out-number conversational controls by at least ten to one, means the well versed programmer must have a good understanding of how manual CNC programs are prepared.

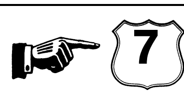
Second, even conventional CNC controls need not be programmed manually. There are many excellent Computer Aided Manufacturing (CAM) systems available to help the manual programmer develop programs. These CAM systems eliminate much of the drudgery of manual programming. However, the experienced programmer must be well acquainted with manual programming techniques even if he or she is using a CAM system to develop programs. There are many times when a good working knowledge of manual programming will help, especially when CAM generated CNC programs must be modified at the machine when the program is to be run. The better the programmer's understanding of manual programming techniques, the faster and better they will be at correcting mistakes and optimizing programs during the programs verification. Without a good knowledge of manual programming the CAM program must go back to the computer to fix even simple mistakes. This wastes precious machine time.

Third, CNC as we know it today has evolved over thirty years of development and constant change. The machine tool and control builders have constantly strived to offer new features that make the usage of CNC equipment easier to work with and more cost effective. For example, many techniques available today were not even possible as little as two years ago. Many of these improvements have been related to how the manual program is developed. Older NC and CNC controls were quite inflexible. The program's format had to be just so. Newer CNC controls are much more forgiving with regard to how the program must be prepared. Also, many of the special techniques currently allowed in manual programming rival even a good CAM system or conversational control. For this reason, and because this evolution is still occurring, it is wise for the beginner to understand manual programming. With these programming enhancements, and especially with simpler applications, it is getting to the point that an experienced manual programmer can outperform the CAM system programmer for simple applications and when a limited number of machines are involved.

1.10.2. Conclusion to key concept number one

Admittedly, this has been a very lengthy key concept. We have been trying to lay some good ground work for what is to come. Hopefully, you are catching on nicely. However, if you are at all confused, we urge you to go back and review this key information. As the course continues, we will be building upon this basic information and assuming you understand the information previously presented.

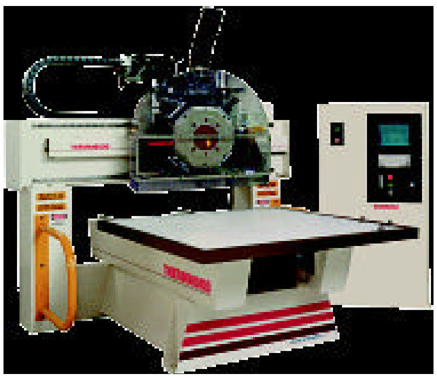
STOP!! Do practice exercise number:



Programming and Operating CNC Routers

and woodworking machining centers

Workbook



Con- tents:

Published By:

CNC CONCEPTS, INC.

Written by Mike Lynch

Document number: S000042