Lesson Plans

Turning Center Programming, Setup, and Operation

FANUC Certified Education - CNC Training

Key Concept 1

Lesson Plan: Know Your Machine from a Programmer's Viewpoint

Introduces course content, Key Concepts approach and Key Concept 1

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine configurations
- 1.2: Turning center speeds and feeds
- 1.3: CNC job work flow
- 1.4: Visualizing the execution of a program
- 1.5: Understanding the workpiece coordinate system
- 1.6: Determining geometry offsets
- 1.7: Setting geometry offsets
- 1.8: Introduction to programming words

2: You Must Prepare to Write Programs
- 2.1: Preparation for programming

3: Understand the motion types
- 3.1: Programming the three basic motion types

4: Know the compensation types
- 4.1: Introduction to compensation
- 4.2: Geometry offsets
- 4.3: Wear offsets
- 4.4: Tool nose radius compensation

5: You must provide structure to your CNC programs
- 5.1: Introduction to program structure
- 5.2: Structured program format

6: Special features that help with programming
- 6.1: One-pass canned cycles
- 6.2: Rough and finish turning and boring (G71 and G70)
- 6.3: Other multiple repetitive cycles (G72-G75)
- 6.4: Threading multiple repetitive cycle (G76)
- 6.5: Sub-programming techniques
- 6.6: Control model differences
- 6.7: Other special programming features

7: Know your machine from a setup person or operator's viewpoint
- 7.1: Tasks related to setup and running production
- 7.2: Buttons and switches on the operation panels

8: Know the three basic modes of operation
- 8.1: The three modes of operation

9: Understand the importance of procedures
- 9.1: The key operation procedures

10: You must know how to safely verify programs
- 10.1: Program verification

Getting started

These lesson plans show you what is being presented in the presentation and reading materials for each lesson. Use them to see what students are learning as they go through the FANUC Certified Education CNC training.

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well. As you can see, there are 10 Key Concepts further divided into 28 lessons. We recommend that you describe the course content, introducing students to the material they will be learning.

Why we're using the Key Concepts approach

- It limits the number of main principles (to 10) a student must understand to become proficient with CNC turning centers.
- It lets students understand precisely where they stand as they go through the class.
- It provides a very good way to organize their thoughts about CNC.
- It provides a building-blocks approach to learning the material. You'll constantly be working from what student's know to what they don't.
- It puts a light at the end of the tunnel.

Programming is explained first

- The first 6 Key Concepts are related to programming. The last 4 are related to setup and operation.
- Many setup- and operation-related topics are discussed in detail during discussions about programming. A programmer must know enough about making setups and running production to direct setup people and operators.
- By the time students get to Key Concept number seven, they will have a very good understanding of many principles needed to setup and run CNC turning centers.

Lessons for this Key Concept

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine configurations
- 1.2: Turning center speeds and feeds
- 1.3: CNC job work flow
- 1.4: Visualizing the execution of a program
- 1.5: Understanding the workpiece coordinate system
- 1.6: Determining geometry offsets
- 1.7: Setting geometry offsets
- 1.8: Introduction to programming words

Key Concept one objective: To ensure that students understand the things a programmer must know about the CNC machine tool they will be working with.

- Students must understand these early lessons. We will be constantly building upon previously presented information.
- These are the things a programmer must know about the machine.
- In Key Concept number seven, students will be learning things a setup person or operator must know about the machine.

Notes:

Machine Configurations

Lesson Plan

1.1

Explains machine components, directions of motion, and programmable functions.

If supplementing the on-line content with lectures:
Explain that the "Getting Started" presentation helps students get familiar with the activities related to each lesson, including the presentation, the reading materials, the test, and, if applicable, the programming activity. In the reading materials for this lesson, students will be introduced to Key Concept 1. You can include this material in a lecture, and briefly describe the lessons for Key Concept number one.

Lesson objective:

To introduce students to the kinds of CNC turning centers that they will be working with.

- We begin by briefly introducing the main topics.
- To skip topics, students can click the topic they want to view.
- They can use the same techniques to review topics.

Main topics for this lesson:

- Basic machining practice
- Machine components
  - Universal slant bed
  - Chucker
  - Twin spindle
  - Gang style
- Directions of motion
  - X axis
  - Z axis
- Programmable functions
  - Spindle
  - Feedrate
  - Coolant
  - Turret
  - Others

Key points made for each topic:

Basic machining practice
- Though beyond the scope for this class, basic machining practice is the key to mastering CNC usage.
- CNC people must understand the basic machining practices related to the CNC machine type being used.
- This understanding must include machining operations (rough and finish turning, rough and finish boring, rough and finish facing, treading, necking, and drilling). They must also understand the processing (sequence of machining operations) used to machine a workpiece.

Machine components
- By showing the main components for those machine types you will be teaching, students will know what makes up a CNC machine tool.
- While students don't have to be machine designers, they should at least be able to properly reference key components by name.
- We show the difference among the various types of turning centers (universal style slant bed, chucker type, vertical type, twin spindle, gang type, etc.).

Directions of motion
- We show the directions of motion (axes) for each kind of turning center you will be teaching.
- We point out that with many machines, the cutting tool (and turret) move along with each axis. This makes it easy to determine and remember the polarity for each axis..
- We explain the polarity (plus versus minus) of each axis – X minus is toward spindle center in X, Z minus is toward the headstock in Z. Directions of motion and polarity are described during each machine type. This makes it easy to display only the machine/s you'll be teaching.

Programmable functions
- We explain that CNC programmers must know the functions of their CNC machine/s that are programmable. Presentations include the four most common programmable functions – spindle, feedrate, coolant, and turret indexing. If your machines have more programmable functions (like a tailstock), be sure to demonstrate them.
- While this presentation includes an introductions to the related programming words, we point out that students need not try to memorize them.

Spindle - we explain that:
- most turning centers allow the spindle to be programmed in three ways, speed (with S), activation (with M03, M04, and M05), and range (with two M codes).
- speed can be specified in revolutions per minute (rpm) or in surface feet per minute (sfm) [which is meters per minute in metric mode].
- M03 (forward) is used for right-hand tools and M04 (reverse) is used for left-hand tools.
- range selection is done with M codes. While M codes vary, M41 is often used for low range and M42 is often used for high range. More about speed and feed control is presented in1.2.

Feedrate – we explain that…
- an F word is used to specify feedrate and feedrate. Feedrate is specified in either per-minute or per-revolution fashion –selected by two G codes (G98 and G99).

Coolant – we explain that…
- coolant is used to cool and lubricate the turning operation. M08 turns on flood coolant and M09 turns it off.

Turret indexing – we explain that…
- most turning centers have a turret – a four-digit T word is used to specify a turret index.

Lesson Plan 1.1 (continued)

At the machine (about 15-minutes)

Once students have completed the reading materials and presentation for this lesson, demonstrate the points made at the machine in your lab. Show the main components and the directions of motion (axes) – be sure to show the polarity for each axis.

Be sure to demonstrate the programmable functions on the machine. Show how to start and stop the spindle, how to move the axes with jog and with the handwheel, show the activation of coolant and index the turret.

Lab exercise

It's never too early to get students touching the machine – but be careful. At this early stage in the class, be sure to provide step-by-step procedures for *anything* you want them to do on the machine – and be sure to watch them carefully when they are practicing.

If you have the FANUC Certification CNC Cart, made by Levil, have them use the operation handbook provided in the Getting Started folder of the on-line content. If you don't, use this operation handbook as a template to create similar procedures for the FANUC-controlled machines you do have in your lab.

Suggested procedures for hands-on practice:

- To start the machine
- To do a zero return

Time to complete lesson 1.1, students must:

- view the on-line presentation for lesson 1.1 (12 minutes)
- read the reading materials for lesson 1.1 (20 minutes)
- take the quiz at the end of the reading materials (5 minutes)

Exercise for lesson 1.1

- Have students take the on-line test for lesson 1.1. (10 minutes)

Approximate total study time for lesson 1.1: **47 minutes**

Notes:

Turning Center Speeds and Feeds

Lesson Plan

1.2

Explain the two ways (each) that speed and feed can be selected.

Lessons in Key Concept 1 (you are here)

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine configurations
- **1.2: Turning center speeds and feeds**
- 1.3: CNC job work flow
- 1.4: Visualizing the execution of a program
- 1.5: Understanding the workpiece coordinate system
- 1.6: Determining geometry offsets
- 1.7: Setting geometry offsets
- 1.8: Introduction to programming words

If supplementing the on-line content with lectures:

Whenever having students start a new lesson, quickly review what has been previously done. Quickly review lesson one: the importance of basic machining practice, machine components, directions of motion, and programmable features. Also be sure to solicit questions about previously presented topics.

Lesson objective:

Help students understand speed and feed selection – as well as which speed and feed mode is most appropriate for a given machining operation.

- Introduce tasks included in this lesson.
- Point out that a manual machinist has only one way to specify speed (rpm) and one way to specify feedrate (per revolution). With CNC turning centers there are *two ways* to specify each.

Main topics for this lesson:

- The two ways to control spindle speed
  - Calculating RPM
  - Constant surface speed
  - Benefits and limitations of constant surfaces peed
- The two ways to control federate
  - Per-revolution
  - Per minute

Key points made for each topic:

Two ways to control spindle speed
- We begin by showing how a manual engine lathe operator must select speed – in rpm. We show the formula for rpm calculation. This calculation is based upon the workpiece diameter being machined. Point out that for many turning operations, like facing and grooving, the diameter being machined constantly changes during the operation – making it difficult (and somewhat infeasible) to specify speed in rpm.
- We then introduce constant surface speed (css), a feature that lets the programmer specify speed directly in surface feet per minute (sfm) if working in the inch mode or meters per minute (mpm) if working in the metric mode.
- We point out that constant surface speed is used for machining operations that require different diameters to be machined (rough and finish turning, rough and finish boring, rough and finish facing, necking, etc.).
- We explain that there are at least two times when the rpm mode must be used – center cutting operations (like drilling) and threading. Also if the turning center is equipped with live tooling, it can only be programmed in rpm mode.
- While it is not of immediate importance, we do show the cycle time limitation of using the constant surface speed mode.
- We explain the two G codes used to specify the speed modes (G96 for constant surface speed mode and G97 for rpm mode).
- We these words in context with a few examples.

The two ways to specify feedrate
- We explain that as with speed, there are two ways to specify feedrate – in per-minute fashion (G98) and in per-revolution fashion (G99). The slide show helps you illustrate per-revolution mode.
- In each feedrate mode, either inches (ipr) or millimeters (mmpr) can be specified. (G20 specifies inch mode, G21 specifies metric mode.)
- Explain that for almost all machining operations, the per-revolution feedrate specification is desirable.
- We point ou that about the only time that feed-per-minute mode should be used is when you want to make a feedrate motion with the spindle stopped. The slide show helps you illustrate one time this is required – when bar feeding.
- If the machine is equipped with live tooling, feedrate must be specified in feed-per-minute mode for any live tooling operation.

At the machine (about 15-minutes)

Run a program that demonstrates the use of constant surface speed mode (a facing operation is ideal). Let students see and hear the spindle accelerate as the cutting tool approaches the center of the spindle (X0). Point out that at center, the spindle will be running at its maximum speed (in rpm) in the selected spindle range.

While it may be a little advanced for this early point in the class, try to show the cycle time limitation of constant surface speed. With a multi- tool Program, let students see what happens as the turret retracts in X to a large-diameter turret index position (the spindle slow down takes longer than the retract motion).

The same will happen when the next tool approaches to a small diameter – there will be a slight delay while the spindle accelerates up to speed. (Much later in the class, you'll be showing how to counteract this limitation.)

Lesson Plan 1.2 (continued)

Lab exercise

Though we have no specific suggestions related to this lesson content, you can have your students continue practicing with procedures needed to run the machine. Again, be very careful, and watch them closely as they run the machine.

Suggested procedures for hands-on practice:

- To start the spindle
- To jog the axes using continuous jog

Time to complete lesson 1.2, students must:

- view the on-line presentation for lesson 1.2 (10 minutes)
- read the reading materials for lesson 1.2. (15 minutes)
- take the quiz at the end of the reading materials (5 minutes)

Exercise for lesson 1.2

- Have students take the on-line test for lesson 1.2 (10 minutes)

Approximate total study time for lesson 1.2: **40 minutes**

Notes:

CNC Job Work Flow

Lesson Plan

1.3

Explains CNC-using company types and tasks related to using a CNC turning center.

Lessons in Key Concept 1 (you are here)

1: Know Your Machine from a Programmer's Viewpoint

- 1.1: Machine configurations
- 1.2: Turning center speeds and feeds
- **1.3: CNC job work flow**
- 1.4: Visualizing the execution of a program
- 1.5: Understanding the workpiece coordinate system
- 1.6: Determining geometry offsets
- 1.7: Setting geometry offsets
- 1.8: Introduction to programming words

If supplementing the on-line content with lectures:

Whenever having students start a new lesson, quickly review what has been previously done. Quickly review lesson one: the importance of basic machining practice, machine components, directions of motion, and programmable features. Also be sure to solicit questions about previously presented topics.

Lesson objective:

To introduce students to the tasks involved with getting a job up and running on a CNC turning center.

We point out that it really helps to understand where CNC machine tools fit into the "bigger picture" of a company's manufacturing environment. CNC programming is but one small part of the picture.

- We show the three most basic company types that use CNC metal cutting machines.
- We simply introduce the tasks related to getting a job up and running on a CNC machine tool. Future lessons will elaborate on these tasks.

Main topics for this lesson:

- Understand the big picture
  - 3 company types
- What will you be doing?
- Flow of programming process
  - Study the print
  - Decide which machine
  - Determine the process
  - Choose tooling
  - Write program
  - Develop documentation
  - Load program

We point out that it really helps to understand where CNC machine tools fit into the "bigger picture" of a company's manufacturing environment. CNC programming is but one small part of the picture.

- We show the three most basic company types that use CNC metal cutting machines.
- We simply introduce the tasks related to getting a job up and running on a CNC machine tool. Future lessons will elaborate on these tasks.

Key points made for each topic:

Understand the big picture

- Different CNC-using companies expect different things from their CNC people.
- The most important factor contributing to personnel utilization is company type.
- The four most basic company types are product producing companies, workpiece-producing companies, tooling-producing companies, and prototype-producing companies.

What will you be doing?

- Students must understand what will be expected of them once they go to work for a CNC-using company..

Flow of the programming process

- We show students the various tasks that must be completed in order to complete a job on a CNC turning center.
- While explaining each task, we point out how many of these tasks require an understanding of basic machining practices.

At the machine (about 15-minutes)

If you have a job up and running on the machine, go out to the machine and point out as much as you can about the various things that must be done prior to running a job (making the work holding setup, the assignment of program zero, the cutting tools, the offsets related to cutting tools, program loading, the verification of the CNC program, etc.).

If you haven't already, run a workpiece to let students see the machine in action. Again, make sure students understand that developing a CNC program is but a small part of the entire CNC process.

Lesson Plan 1.3 (continued)

Lab exercise

Though we have no specific suggestions related to this lesson content, you can have your students continue practicing with procedures needed to run the machine. Again, be very careful, and watch them closely as they run the machine.	Suggested procedures for hands-on practice: ▪ To jog axes using incremental jog ▪ To use the handwheel
Time to complete lesson 1.3, students must: ▪ view the on-line presentation for lesson 1.2 (9 minutes) ▪ read the reading materials for lesson 1.2. (8 minutes) ▪ take the quiz at the end of the reading materials (5 minutes)	**Exercise for lesson 1.3** ▪ Have students take the on-line test for lesson 1.2 (10 minutes) Approximate total study time for lesson 1.3: **32 minutes**

Notes:

Visualizing Program Execution

Lesson Plan

1.4

Explains the importance of being able to visualize the movements of cutting tools.

Lessons in Key Concept 1 (you are here)

1: Know Your Machine from a Programmer's Viewpoint

- 1.1: Machine configurations
- 1.2: Turning center speeds and feeds
- 1.3: CNC job work flow
- **1.4: Visualizing the execution of a program**
- 1.5: Understanding the workpiece coordinate system
- 1.6: Determining geometry offsets
- 1.7: Setting geometry offsets
- 1.8: Introduction to programming words

If supplementing the on-line content with lectures:

Solicit questions about previous topics. Quickly review machine configurations, direction of motion, programmable features. Remind students that there are many tasks that must be completed in order to get a CNC machine up and running.

Point out that a programmer must be able to "see" the program's execution in their mind.

Lesson objective:

To get students to understand the importance of visualizing the program's execution. Without this ability, *they cannot write programs*.

- Shows an analogy related to providing travel instructions.
- Students will see their first complete program in this lesson.
- Provides several points about program structure.

Main topics for this lesson:

- Sequential order of program execution
- Program make-up
- Visualizing a program's movements

This lesson begins by reminding students about the importance of understanding basic machining practices. A machinist has seen many machining operations taking place. When writing a program, a CNC programmer must "see" the program's execution in their mind while sitting at a bench or desk.

Key points made for each topic:

Sequential order of program execution – we explain that:

- writing a CNC program is like giving any other set of step-by-step instructions.
- programs are made up of commands – commands are made up of words – words are made up of a letter address (N, G, X, Z, etc.) and a numerical value.
- programs are executed sequentially. The machine will read, interpret the first command in the program. Then it will go on to the next command. Read, interpret, execute. It will continue in this fashion for the balance of the program.
- Visualizing program executions – students must understand that:
- without the ability to visualize a program's execution, they cannot write CNC programs.
- even experienced machinists can have problems visualizing program execution.
- an understanding of how machining operations are performed is necessary to visualize.
- just as you cannot create a set of travel instructions without being able to visualize the path (a nice analogy is provided in the slide presentation), neither can you write a CNC program without being able to visualize how cutting tools will move through their paths.
- a machinist has everything needed to complete the job right in front of them (machine, work holding device, cutting tools, etc.). The programmer must write a program while sitting behind a desk, armed with only a print and a calculator.

Program make-up

- We explain commands are made up of words, and that words are made up of a letter address and a numerical value (number).

An example job (machinist versus programmer)

- We show a simple example job – first done by a machinist, then by a CNC program. This is the first complete program a student will see. While showing the program, we explain each line, making sure students understand that they don't have to memorize the related commands.
- We stress that programs will be executed sequentially (just like a person following a set of travel instructions).
- We stress the general make-up of commands and words in the program.
- Most importantly, we stress the importance of visualization – if the programmer cannot "see" the machining operation in their mind, they cannot write the program.

Lesson Plan 1.4 (continued)

At the machine (about 15-minutes)

If students are at all weak in their basic machining practice skills, take them out to a machine and demonstrate the motions of the most common machining operations, including drilling, tapping, reaming, face milling, and side cutting.

Admittedly, this may not be enough to get them comfortable with the related operations, but at least they'll know what each cutting tool is designed to do. Again, basic machining practice experiences is a prerequisite for this course.

Lab exercise

Though we have no specific suggestions related to this lesson content, you can have your students continue practicing with procedures needed to run the machine. Again, be very careful, and watch them closely as they run the machine.

Suggested procedures for hands-on practice:

- To set axis displays (origin)
- To set axis displays (preset)

Have students take a simple measurement, jogging an axis to one position, presetting an axis display, and then jogging to another position (as is done with tool length measurements)

Time to complete lesson 1.4, students must:

- view the on-line presentation for lesson 1.4 (9 minutes)
- read the reading materials for lesson 1.4. (10 minutes)
- take the quiz at the end of the reading materials (5 minutes)

Exercise for lesson 1.4

- Have students take the on-line test for lesson 1.4 (10 minutes)

Approximate total study time for lesson 1.4: **34 minutes**

Notes:

Understanding the Workpiece Coordinate System

Lesson Plan

1.5

Explains how programmed coordinates are determined.

If supplementing the on-line content with lectures:
Solicit questions about previous topics. Review the tasks related to getting a machine up and running. Review the importance of being able to visualize the execution of a CNC program.
Point out that a programmer must be able to determine positions through which cutting tools will move. Have students remember the program shown in lesson three. In this program, a drill is commanded to move through certain positions so that it could machine a hole. In this lesson, we'll be showing how to determine tool path positions.

Lesson objective:

To show students how positions (coordinates) are determined for use within a program. We explain the absolute mode – and that all positions used in a program are specified from a common location (the program zero point).

- Students will be determining positions in a two dimensional coordinate system from a central origin.

Main topics for this lesson:

- The workpiece coordinate system
- Graph analogy
- Placing program zero
- Right versus left end of workpiece
- Calculating coordinates
- Absolute versus incremental positioning

Key points made for each topic:

The workpiece coordinate system

- We begin with an explanation of how an axis drive system works. While students don't need to know all of the inner workings of a turning center, the point we're making has to do with how positions are specified within a program. The question we eventually ask is: *How many rotations of a drive motor equate to 1" of linear motion?* Because of the workpiece coordinate system, programmers need not know the answer.

Graph analogy

- Next, we show an analogy related to making a graph. The graph in the presentation is for a company's productivity. We relate each component of a graph to the related components of the workpiece coordinate system as it is used for CNC turning centers.
- We point out that, in CNC terms, the origin of the workpiece coordinate system is called the program zero point. All coordinates used within a CNC program will be specified from this point. The graph analogy shows a two-dimensional coordinate system (XZ).

How coordinates are calculated

- While with the graph, all positions are plotted up and to the right of the origin (quadrant number one), with CNC coordinate systems, a tool must often move to a position that is to the left of or below the program zero point. We point out that every coordinate used in a CNC program has a polarity (plus or minus).
- With coordinates having a positive polarity, the polarity sign (plus) is assumed. Students must only include a polarity sign with negative coordinates (-).
- We make sure students understand that X coordinates specify a diameter. Some dimensions (like chamfers are specified in radial fashion.

Where to place the program zero point – we point out that:

- the wise placement of program zero will minimize the number of calculations needed to determine coordinates for the program.
- the program zero point is determined based upon print dimensioning. The datum surfaces for the drawing will be the program zero point surfaces for the program.
- these will be the same surfaces used for workpiece location in the work holding setup.
- the program zero in the X axis is always the spindle/workpiece center. Program zero in Z is usually placed at the right end of the finished workpiece. (All examples in the slide show and student manual use this position as program zero.)

Absolute versus Incremental positioning - Points made:

- When coordinates are specified from program zero, it is called the absolute mode of programming.
- With most turning centers, X and Z specify absolute positions.
- Students should concentrate on absolute positioning.
- Another positioning mode is available: the incremental positioning mode. With this positioning method, positions are commanded from the cutting tool's current position.
- With most turning centers, U and W are used to specify incremental departures. U specifies a diameter increase or decrease, W specifies an incremental departure distance in the Z axis.
- Programs written incrementally are difficult to follow.
- If a mistake is made in a series of incremental positions, every movement from the point of the mistake will be incorrect.

Lesson Plan 1.5 (continued)

At the machine (about 20-minutes)

In you lab, run a program on the machine. You don't have to cut anything, but it might help hold attention if you do. As the program runs, monitor the **absolute** position display screen on the control. This screen, of course, constantly shows position relative to the program zero point.

Based upon watching this screen as the program executes, see if anyone can determine the program zero point position for the program. You might also want to introduce the other display screen pages (relative, machine, and distance-to-go).

Lab exercise (about 5-minutes per student)

We have no suggestions for lab exercises for this lesson. Students can continue practicing with procedures.

Suggested procedures for hands-on practice:
- To enter wear offsets

Time to complete lesson 1.5, students must:
- view the on-line presentation for lesson 1.5 (13 minutes)
- read the reading materials for lesson 1.5. (15 minutes)
- Fill in the coordinate sheet in the reading materials - p7. (5 minutes)

Exercises for lesson 1.5
- Have students take the on-line test for lesson 1.5 (10 minutes)
- Have students complete and submit the coordinate sheet exercise for lesson 1.5 (10 minutes)

Approximate total study time for lesson 1.4: **53 minutes**

Notes:

Determining Geometry Offsets

Lesson Plan 1.6

Explains how the values needed to assign program zero are determined.

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Explain that while this lesson is more related to setup and operation (especially setup), programmers must know enough about making setups to direct setup people (providing the appropriate documentation). For this reason, programmers must know how program zero is assigned at the machine.

Lesson objective:

Ensure that students understand that program zero must be *assigned* – and that to assign program zero, certain values called *program zero assignment values* must be determined.

Main topics for this lesson:

- Why program zero must be assigned for each tool
- Understanding the zero return position
- What program zero assignment values represent

Key points made for each topic:

Why program zero must be assigned for each tool

- We explain that each cutting tool (especially different types of cutting tools) have a different position in the turret. The tip of a turning tool, for example, will not be in the same position as a drill or boring bar.

How program zero assignment values are determined

- We briefly introduce the most common ways to assign program zero. We'll be elaborating on these methods in lesson 1.7.

Zero return position – Students must understand that:

- the zero return position is the point of reference for program zero assignment values.
- the zero return position is a reference position on the machine – commonly placed very close to the plus over-travel limit for each axis.
- the *machine position display screen* shows the machine position relative to the zero return position.
- with some machines, two lights – called *axis origin lights* – will come on when the machine is at its zero return position.

Alternatives for program zero assignment

- *Assigning program zero in the program* – This method is used with old machines that don't have geometry offsets.

Assigning program zero with geometry offsets and the measure function – This method is used if the machine has geometry offsets, but not the work shift function.

- *Assigning program zero with geometry offsets and the work shift function* – This method is the preferred method unless the machine has a tool touch-off probe.
- *Assigning program zero with the tool touch-off probe* – This is the preferred method if the machine has a tool touch-off probe.

Program zero assignment values for center cutting tools

- We illustrate how to determine program zero assignment values for drill, taps, reamers, and other center cutting tools.
- The X axis program zero assignment value will be the same for any drill, tap, or reamer.

Program zero assignment values for boring bars

- We illustrate a simple way of determining the X axis program zero assignment value for any boring bar.

Other considerations about program zero assignment

- In the reading materials, we present a few more points related to trial machining, sizing, and replacing dull cutting tools

Lesson Plan 1.6 (continued)

At the machine (about 20-minutes)

With a workholding setup made on your lab machine, demonstrate the techniques used to measure geometry offset values.

This demonstration will require procedures to start the spindle, jog the axes, use the handwheel, set and reset the relative position displays, and use the measure function. Be sure to use the axis displays to illustrate what geometry offset values represent.

Lab exercise (about 5-minutes per student)

With the specific procedures, have students practice measuring program zero assignment values in the same fashion just demonstrated. Again, be careful to monitor their progress.

Suggested procedures for hands-on practice:

- To measure and enter program zero assignment values

To complete lesson 1.6, students must:

- view the on-line presentation for lesson 1.6 (11 minutes)
- read the reading materials for lesson 1.6 (20 minutes)
- take the quiz at the end of the reading materials (10 minutes)

Exercises

- Have students take the on-line test for lesson 1.6. (10 minutes)
- Have students complete and submit the coordinate sheet exercise for lesson 1.6 (15 minutes)

Approximate total study time for lesson 1.5: **66 minutes**

Notes:

Setting Geometry Offsets

Lesson Plan

1.7

Explains how program zero is actually assigned.

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Explain that this lesson is also more related to setup and operation (especially setup). But again programmers must know enough about making setups to direct setup people (providing the appropriate documentation). For this reason, programmers should know how program zero is assigned at the machine.

Lesson objective:

Be sure students understand how program zero is assigned – with geometry offsets and the work shift value.

Main topics for this lesson:

- Assigning program zero with geometry offsets
    - Advantages
    - Measurement polarity

Key points made for each topic:

Assigning program zero with geometry offsets - we point out that:

- Regardless of how program zero is assigned, the program zero assignment values shown in lesson 1.6 will be used.
- With geometry offsets, the polarity for program zero assignment values is taken *from the zero return position to program zero* (almost always negative).
- The program zero assignment values are entered into the appropriate geometry offset registers.
- If the *work shift value* is used as shown in lesson 1.6, geometry offsets for tools used in the previous job will remain correct.
- The *measure function* can facilitate the measurement and entry of geometry offset values.
- A tool touch-off probe should be the method of program zero assignment if the machine has one.

A few more points about program zero assignment

- We make some points about trial machining and sizing – as well as what must be done when dull tools must be replaced.

Lesson Plan 1.7 (continued)

At the machine (about 20-minutes)

We're assuming you are using geometry offsets to assign program zero (again, you should use geometry offsets to assign program zero unless your machine does not have them). Use the program zero assignment values measured in lesson six and show how they are entered into geometry offset registers.

If your machine has a tool touch-off probe, show its use. If not show how the measure function helps with the entry of geometry offset values. Show the geometry offset pages

If you have an old machine that doesn't have geometry offsets, then show how the G50 commands in a program must be edited in order to assign program zero.

Lab exercise (about 5-minutes per student)

Again, have students use the procedures in the operation handbook to enter geometry offsets so they can practice with minimal help from you. Have them work with an unused workpiece coordinate system offset for practicing (like geometry offset number 32 so they cannot overwrite needed offset values.

Suggested procedures for hands-on practice:

- To measure and enter geometry offsets (repeated)
- To measure and enter work shift value

To complete lesson 1.7, students must:

- view the on-line presentation for this lesson (9 minutes)
- read the reading materials for this lesson (25 minutes)
- fill in the coordinate sheet at the end of the reading materials (10 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete and submit the coordinate sheet exercise for this lesson (15 minutes)

Approximate total study time for this lesson: **69 minutes**

Notes:

Introduction to Programming Words

Lesson Plan

1.8

Introduces students to the word types used in CNC programs.

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Point out that in Key Concept number one, students have been exposed to several programming words. In this lesson, we show all of the word types used in programming

Lesson objective:

To acquaint students with the word types (letter addresses) used in CNC turning center programs.
In this final lesson for Key Concept number one, we explain the meaning of all CNC word types. Though students need not try to memorize every work type, it helps if they can see the limited number of different words available to CNC programmers.

Main topics for this lesson:

- All programming words
  - O, N, G, X, Y, Z, R, Q, S, T, P, M, etc.
- Command limitations

Key points made for each topic:

Introduction to word types

- We point out that there are only about 50-60 different words used in CNC Turning Center Programming. We ask students to look at learning programming as like learning a foreign language that has only 60 words.
- We explain that many word types are easy to remember (like T for tool, S for speed, and F for federate). Others are not so easy to remember (like O for program number and N for sequence number).

Word types

- We present the various word types. In each case, we explain whether the word is a real number (allowing a decimal point) or an integer (whole number). We also specify the format for the word. Finally, we explain the word meaning, including primary and any secondary uses for the word.
- The reading materials for this lesson include a full list of G an M words. We point out that M words are determined by machine tool builders and can vary from machine to machine.

Lesson Plan 1.8 (continued)

At the machine (about 10 minutes)

While it doesn't have to be at the machine tool (the text editor of a computer will work), call up a program and point to the various words. See if students can remember any of the word meanings for words they see in the program. It's likely that they will so praise them for their efforts.

Show the program check display screen page, which shows the currently active CNC words. Again, see if students can remember them.

Lab exercise (about 5-minutes per student)

We have no suggestions for lab exercises that are related to this lesson. If you wish, you can have students continue practicing with machine operation procedures.

To complete lesson 1.8, students must:

- view the on-line presentation for this lesson (14 minutes)
- read the reading materials for this lesson (10 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete and submit the coordinate sheet exercise for this lesson (15 minutes)

Approximate total study time for this lesson: 49 **minutes**

Notes:

Key Concept

2

Lesson Plan: You Must Prepare to Write Programs

Introduces Key Concept number two.

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine configurations
- 1.2: Turning center speeds and feeds
- 1.3: CNC job work flow
- 1.4: Visualizing the execution of a program
- 1.5: Understanding the workpiece coordinate system
- 1.6: Determining geometry offsets
- 1.7: Setting geometry offsets
- 1.8: Introduction to programming words

2: You Must Prepare to Write Programs
- 2.1: Preparation for programming

3: Understand the motion types
- 3.1: Programming the three basic motion types

4: Know the compensation types
- 4.1: Introduction to compensation
- 4.2: Geometry offsets
- 4.3: Wear offsets
- 4.4: Tool nose radius compensation

5: You must provide structure to your CNC programs
- 5.1: Introduction to program structure
- 5.2: Structured program format

6: Special features that help with programming
- 6.1: One-pass canned cycles
- 6.2: Rough and finish turning and boring (G71 and G70)
- 6.3: Other multiple repetitive cycles (G72-G75)
- 6.4: Threading multiple repetitive cycle (G76)
- 6.5: Sub-programming techniques
- 6.6: Control model differences
- 6.7: Other special programming features

7: Know your machine from a setup person or operator's viewpoint
- 7.1: Tasks related to setup and running production
- 7.2: Buttons and switches on the operation panels

8: Know the three basic modes of operation
- 8.1: The three modes of operation

9: Understand the importance of procedures
- 9.1: The key operation procedures

10: You must know how to safely verify programs

10.1: Program verification

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well. As you can see, there are ten Key Concepts further divided into 28 lessons.

Key Concept number two is a short, one-lesson key concept. Though it is short, it is among the most important Key Concepts.

Lessons for this Key Concept

2: You Must Prepare to Write Programs
- 2.1: Preparation for programming

Key Concept two objective: Help students understand the steps that must be taken prior to writing a program.

Notes:

Preparation for Programming

Lesson Plan

2.1

Introduces students to the word types used in CNC programs.

Lessons in Key Concept 2 (you are here)

You Must Prepare to Write Programs

2.1: Preparation for programming

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Explain that even though Key Concept number two has little or nothing to do with programming words and commands, it is among the most important Key Concepts. Programmers must be prepared to write programs. With preparation done, writing a program is simply a matter of translating the plan into a language the CNC turning center can understand.

This class, of course, is presenting G code level, manual programming. However, the preparation steps we show in this lesson are necessary regardless of how programs are prepared. If, for example, students will eventually be using a computer aided manufacturing (CAM) system to prepare programs, all of the preparation steps we show (except doing the math) will be required.

- Remind students that adequate preparation will make programming much simpler, reducing the potential for mistakes. Frankly speaking, the quality of most programs is directly related to the quality of the preparation done before the program is written.

Lesson objective:

To ensure that students understand and can perform the steps required to prepare to write CNC programs.

Main topics for this lesson:

- Divide and conquer
- Preparation and safety
- Typical mistakes
- Preparation steps

Key points made for each topic:

Preparation steps

- We point out that any complex task can be simplified by breaking it into small pieces. In a sense, we're providing a way to divide and conquer.
- We provide an analogy for making a speech. Just as an ill-prepared speaker will be likely to make mistakes during the presentation, so will the ill-prepared programmer be prone to making mistakes.

Study and mark up the print

- We explain that in most companies, the programmer is given a working copy of the workpiece drawing (print). They can mark up this print in any way that helps them understand the job.
- We explain that the programmer should mark up the location of program zero, they should mark the surfaces that get machined, they should draw in chuck jaws and other obstructions, and in general, they should mark up anything that will help them during programming.

Develop the machining process

- We explain a process planning form that is provided in the reading materials.
- We provide the benefits of this form. Before the program is written, the programmer is forced think through: the process and all cutting tools used in the job, possible tooling interference problems, and cutting conditions fore each cutting tools.
- We point out that this completed form is the English-version of the program. Writing the program will be a simple matter of translating this form into a language the CNC turning center can understand.
- We explain that this form also makes great documentation for anyone who must work on the program in the future.

Do the math

- We point out that doing the math up-front will keep the programmer from breaking out of their train of thought when programming to come up with coordinates needed in the program.
- We demonstrate our recommended method of calculating coordinates – numbering each point on the print through which cutting tools will move and making a *coordinate sheet* that has all coordinates for these points. (This should be familiar to students if they have been doing the exercises.)
- We point out that often the coordinates needed in the program are not specified right on the print. Our example shows the milling of a circular pocket.

Plan the setup

- We point out that there are many things about the setup that affect the way a program must be written. For example, jaws and other obstructions must be avoided by cutting tools. For this reason, the programmer must understand how the setup will be made before they can write a program.
- We describe a setup sheet (that is also in the reading materials), helping students understand the things that must be documented for the setup person.

Lesson Plan 2.1 (continued)

At the machine

We have no recommendations for things to do at the machine that are related to this lesson's content.

Lab exercise

We have no suggestions for lab exercises that are related to this lesson. If you wish, you can have students continue practicing with machine operation procedures.

To complete lesson 2.1, students must:

- view the on-line presentation for this lesson (11 minutes)
- read the reading materials for this lesson (25 minutes)
- complete the coordinate sheet activity at the end of the reading materials (15 minutes)
- complete the exercise to develop tool paths at the end of the reading materials (20 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete and submit the coordinate sheet exercise for this lesson (15 minutes)

Approximate total study time for this lesson: **97minutes**

Notes:

Key Concept

3

Lesson Plan: Understand the Motion Types

Introduces Key Concept number three.

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well. As you can see, there are ten Key Concepts further divided into 28 lessons.

Key Concept number three is one-lesson key concept.

Points made during the introduction to Key Concept number three

Interpolation

- We remind students that they currently know how to determine coordinates (positions) through which cutting tools will move (this is presented in lesson 1.5). But they must also know what it takes to command *how a cutting tool will move from point to point*.
- Next, we describe *interpolation*. While students may not need to know the details of how interpolation works, it helps to understand what the machine is doing for them.
- We also briefly introduce the three most common motion types in this session – rapid motion, straight line (linear) motion, and circular motion.

Lessons for this Key Concept

3: Understand the Motion Types
- 3.1: Programming the three basic motion types

Key Concept three objective: Help students understand and master the motion types for CNC turning centers.

Notes:

Programming the Three Basic Motion Types

Lesson Plan

3.1

Presents the three most common ways a CNC turning center can move.

Lessons in Key Concept 3 (you are here)

3: Understand the Motion Types

3.1: Programming the three basic motion types

If supplementing the on-line content with lectures:

Explain that programming the motion types is remarkably simple. Students must specify the kind of motion they want, the end point for the motion, and possibly a feedrate. Circular motions additionally require the arc size to be specified.

- Motion types share several things in common. When students understand one motion type, similar techniques will be used for all.
- With the preparation steps described in lesson eight completed (especially the math), commanding motion should be relatively simple.

Lesson objective:

To bring students to a level that they understand and can command the three most common motion types.

Main topics for this lesson:

- Understanding interpolation
- Motion commonalities
- Cutting tool point that is programmed
- Rapid motion
- Linear motion
- Circular motion

Key points made for each topic:

Interpolation and the three motion types

- We present these topics in the introduction to Key Concept number three.

Motion commonalities

- We present the five things that all motion types share in common: all are modal, the end point is commanded, only moving axes are specified, all are affected by positioning mode (absolute or incremental), and the leading zero for each can be suppressed (G0 is the same as G00, etc.).

Point programmed

- We point out that beginning programmers often have a problem with this. They must understand the actual point of the cutting tool that they are programming.
- For center cutting tools (drills, taps, reamers, etc.), the tool center in X and the tool tip in Z is being programmed.
- For single point cutting tools (like turning tools and boring bars, it is the extreme tip of the tool in X and Z that is being programmed.

Rapid motion – We explain that:

- rapid motion is commanded by G00.
- motion occurs at the machine’s fastest possible rate.
- a straight motion may not occur when two or more axes are specified (one axis will probably reach its destination first).
- rapid motion is used to reduce program execution time (whenever the cutting tool is not cutting, rapid motion should probably be used).
- We show examples to stress these points.

Linear motion – We explain that:

- linear motion (also called straight-line cutting motion) is commanded by G01.
- motion will occur along a straight line, even if more than one axis is specified.
- a feedrate (F word) must be specified in (at least) the first linear motion command. Feedrate is modal – if a series of motions must occur at the same feedrate, only the first motion command requires a feedrate.
- this command is used when machining must occur along a straight line.
- We show several examples to stress these points.

Circular motion – We explain that:

- G02 specifies clockwise circular motion – G03 specifies counter clockwise circular motion.
- motion will occur along a circular path.
- the arc size must be specified (with an R word).
- the current feedrate will be used.
- these commands are used when machining circular surfaces.
- We show several examples to stress these points.

Specifying arc size

- We present the two ways to specify arc size – with an R word to specify arc size directly and with directional vectors (I, J, and K). We recommend that students use the R word. But for the sake of completeness, we do show how directional vectors are used.

Arc limitations

- We present the limitations of circular motion commands. For example, we point out that it is possible to cross only one quadrant line (arc centerline) per command. This means that it is not possible to generate an arc greater than 180 degrees per command. Larger arcs must be broken into two commands.

Lesson Plan 3.1 (continued)

Polar coordinate interpolation

- We introduce a fourth motion type called *polar coordinate interpolation*. This motion type is used with mill/turn machines having a C-axis and live tooling.
- We explain that mill/turn machines are described in the appendix that follows lesson 6.7.

At the machine (20-30 minutes)

Have students work on their first two programs in the activities related to this lesson (in the reading materials and in in the programming activity). You can use either of these programs to help them get some meaningful practice at the machine. Or, if you have developed your own practice program (perhaps that actually machines a workpiece), you can use it instead.

Have them type the program into the control (meaning you'll need a step-by-step procedure that shows them how to enter new programs). Be sure to double check this program for mistakes since you're not going to be teaching program verification techniques at this point. You may elect to let them practice typing the program into the control – but use your own proven program (that you *have* verified) when you actually run the program.

This also makes a great time to quiz them on some of the tasks that must be done prior to running a program. From what students should know so far, they should quickly point out that program zero must be assigned. Review the techniques for measuring program zero assignment values and entering them into geometry offsets.

When you're ready, run the program for them. Again, it might be wise to simply air cut – without a workpiece in position. Students can still nicely see the program's execution, including the three motion types introduced in this lesson.

Lab exercise

We have no recommendations for lab exercises related to this lesson. Since students commonly work on their own when doing lab exercises, we don't recommend letting them run their own programs (yet).

To complete lesson 3.1, students must:

- view the on-line presentation for this lesson (13 minutes)
- read the reading materials for this lesson (25 minutes)
- complete the programming activity at the end of the reading materials (15 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (25 minutes)
  - If using NC Guide, have students type the program with NC Guide and verify the program (20 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: **108 minutes**

Notes:

Key Concept 4	***Lesson Plan:*** ***Know the Compensation Types***
	Introduces Key Concept number four.

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well. As you can see, there are ten Key Concepts further divided into 28 lessons.

Key Concept number four is four-lesson key concept.

Points made during the introduction to Key Concept number four

- Key Concept number four contains four lessons that are related to certain unpredictable variables of tooling (work holding tools and cutting tools).
- We point out that a programmer won't know every detail about a setup is made as the program is being written. These compensation types allow the programmer to ignore certain tooling-related information while they write the program.
- We explain that later (commonly at the machine during setup), the setup person (or someone) will determine and enter this information into the machine separate from the program.

Lessons for this Key Concept

4: Know the compensation types
- 4.1: Introduction to compensation
- 4.2: Geometry offsets
- 4.3: Wear offsets
- 4.4: Tool nose radius compensation

Key Concept four objective: Help students understand and master the three compensation types.

Notes:

Introduction to Compensation

Lesson Plan

4.1

Explains why compensation is required with CNC turning centers.

Lessons in Key Concept 4 (you are here)

4: Know the compensation types
- **4.1: Introduction to compensation**
- 4.2: Geometry offsets
- 4.3: Wear offsets
- 4.4: Tool nose radius compensation

If supplementing the on-line content with lectures:
Explain that there are certain things that a programmer won't know about the machine's setup while a program is written. Compensation lets the programmer write the program without knowing these things. For example, the programmer won't know the physical position of program zero in the setup.

Lesson objective:

To make sure that students understand why compensation types are required.

Main topics for this lesson:

- Marksman analogy
- Understanding compensation
  - Geometry offsets
  - Wear offsets
  - Tool nose radius compensation
- Trial machining

Key points made for each topic:

Analogies
- We show a marksman analogy in the reading materials and presentation. It is remarkably similar to how the compensation types are used on CNC turning centers. It should help students understand that an initial compensation setting may not be perfect. The tighter the tolerance, the more likely it will be that a second adjustment will be necessary after a cutting tool machines.
- We explain that if tolerances are small, it may be necessary to make an initial adjustment that forces the cutting tool to leave excess stock. After machining, another adjustment will be necessary.

More on tolerances
- The reading materials include a presentation about tolerance interpretation during the introduction to Key Concept number four.
  - We use this to ensure that students understand tolerance bands, whether a measured dimension is within the tolerance band, the target value for an adjustment, and how much adjustment (the deviation) is required.

Understanding offsets
- From the marksman analogy, we point out that offset settings are like the amount of sight adjustment needed for the rifle.
- We also compare CNC offsets to the memories of an electronic calculator – they are referenced by a number and they have no meaning until they are invoked. But unlike calculator memories, CNC offsets are more permanent. They are retained after the machine's power is turned off.
- We explain that it is in offset registers that students will be entering certain tooling related information (like program zero assignment values, values related to tool wear, and values related to the radius on the cutting edge of single point cutting tools.
- We describe the various offset display screen pages on a typical CNC turning center.

Trial machining
- While more detailed descriptions of trial machining are shown during the lessons for each compensation type, we introduce trial machining in this lesson.
- We show an example workpiece that has close enough tolerances to require trial machining

Lesson Plan 4.1 (continued)

At the machine (10 minutes)

Show students the various offset pages for the machine/s they will be working on. Demonstrate how offsets are entered (again provide a step-by-step procedure for entering offsets).

Reiterate the importance of knowing how to determine offset adjustments. Anyone can follow the procedure to enter an offset value. It takes more of an understanding to know *the value* to enter.

Lab exercise

We have no suggestions for lab exercises that are related to this lesson

To complete lesson 4.1, students must:

- view the on-line presentation for this lesson (9 minutes)
- read the reading materials for this lesson (15 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (25 minutes)
  - If using NC Guide, have students type the program with NC Guide and verify the program (20 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: **79 minutes**

Notes:

Geometry Offsets

Lesson Plan

4.2

Extends what students know about geometry offsets from lessons 1.6 and 1.7.

Lessons in Key Concept 4 (you are here)

4: Know the compensation types

- 4.1: Introduction to compensation
- **4.2: Geometry offsets**
- 4.3: Wear offsets
- 4.4: Tool nose radius compensation

If supplementing the on-line content with lectures:

Most of what students need to know about geometry offsets is presented in lessons six and seven. Indeed, if program zero assignment values are measured at the machine (which is commonly the case), it is possible to skip this lesson if you're running short on time.

Lesson objective:

To complete students' understanding of geometry offsets.

Main topics for this lesson:

- Why geometry offsets are required
- Review of program zero assignment
- Work shift value
- Minimizing work from job to job

Key points made for each topic:

Why geometry offsets are required

- We show the benefits of geometry offsets over assigning program zero in the program with G50.

Review of program zero

- We review the assignment of program zero.

Understanding work shift

- Again, work shift is described in lesson seven. What is included in this lesson is review.

How accurate are program zero assignment measurements?

- This presentation reiterates the importance of trial machining.
- The reading material includes a lengthy explanation of the actual workings of geometry offsets and work shift.

Lesson Plan 4.2 (continued)

At the machine (about 10 minutes)

Show students the various offset pages for the machine/s they will be working on. Demonstrate how offsets are entered (again provide a step-by-step procedure for entering offsets).

- Reiterate the importance of knowing how to determine offset adjustments. Anyone can follow the procedure to enter an offset value. It takes more of an understanding to know *the value* to enter.

Lab exercise

Since it is such an important skill, have students continue practicing the measuring and entry of geometry offset ad work shift values.

Suggested procedures for hands-on practice:

- To measure and enter geometry offsets (repeated)
- To measure and enter work shift value (repeated)

To complete lesson 4.2, students must:

- view the on-line presentation for this lesson (7 minutes)
- read the reading materials for this lesson (10 minutes)
- do the programming activity at the end of the reading materials (20 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (25 minutes)
  - If using NC Guide, have students type the program with NC Guide and verify the program (20 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: **92 minutes**

Notes:

Wear Offsets

Lesson Plan

4.3

Explains how wear offsets can be used for trial machining and sizing.

Lessons in Key Concept 4 (you are here)

4: Know the compensation types

- 4.1: Introduction to compensation
- 4.2: Geometry offsets
- **4.3: Wear offsets**
- 4.4: Tool nose radius compensation

If supplementing the on-line content with lectures:

Point out that wear offsets allow the programmer to write programs without knowing how much tool wear will be experienced during each tool's life. These offsets can also be used when the setup person is worried about whether the initial setting of the geometry offset (program zero assignment) is accurate enough.

Point out that the tolerances held by today's turning centers are very small. It is not uncommon for tuning center users to hold overall tolerances of under 0.001 inch on a regular basis. For diameter dimensions, if a tool's program zero assignment is off by 0.0005 inch, the resulting diameter deviation will be 0.001 inch.

- Even if geometry offsets are measured and entered *perfectly*, there can still be a substantial deviation when cutting the first workpiece (tool pressure causes this deviation).
- Wear offsets can be used for trial machining when there is a doubt about the accuracy of the initial geometry offset setting.
- As a tool dulls a small amount of material will be removed from its cutting edge. With small tolerances, this wear may cause the dimension to grow or shrink out of its tolerance band.

Lesson objective:

To help students master the understanding and use of wear offsets.

Main topics for this lesson:

- Why wear offsets are required
- How wear offsets work
- Transitioning from job to job
- When to clear

Key points made for each topic:

Why wear offsets are required

- We present five reasons why wear offsets are required.

How wear offsets work

- This topic in the presentation begins by helping you explain how a "total" offset is determined. The control will total the wear offset, the geometry offset and the work shift value to determine the total amount of offset used for program zero assignment.
- We provide some example scenarios for machining with a tool to illustrate how wear offsets work.

More on trial machining

- We point out that while wear offsets can be used for trial machining, it might be better to use geometry offsets. If this is done, the wear offset for each tool can begin the production run at zero. When a dull tool is replaced, the operator will easily know how to reset the wear offset (to zero) after replacing the tool.
- This, of course, is only important with large production runs when tools will need replacing during the production run.

Transitioning from one job to the next

- We explain that when a tool is machining properly in one job, it will continue to do so in the next – especially if the work shift value is being used. A common mistake is clearing offsets when a job is finished.
- We point out that offsets should only be cleared (set to zero) when the related tool is removed from the turret. In this way, the setup person and operator can rest assured that if a cutting tool is in the turret (from any previous job), the related offsets are still correct.
- *Special point:* When an adjustment is necessary, students must understand, of course, the amount by which an offset must be changed. An explanation of determining the deviation from the measured value to the target value is provided in the introduction to Key Concept number four.

Lesson Plan 4.3 (continued)

At the machine (about 15 minutes)

- With a simple program (we recommend simply turning one diameter), demonstrate wear offset adjustments. If actually cutting a workpiece, you can show how a small modification to the wear offset can make the cutting tool remove more material.
- Quiz students about which way wear (or geometry) offsets must be changed (plus or minus) in order to leave excess stock for trial machining.
- Show students how to modify the current values within offsets (commonly by using the +INPUT function). With this function, point out that students must only calculate the deviation – not the new offset value

Lab exercise

With a few sample workpieces available for measuring, provide students with a dimension and tolerance for some "critical" workpiece attributes. Be sure students can come up with the target value (commonly the mean value). After taking physical measurements on the workpieces, have them specify the amount of deviation (as well as polarity) for offset adjustment.

Suggested procedures for hands-on practice:

- To enter wear offsets

Go through a few scenarios, providing realistic examples of what could happen while running the first workpiece and during a production urn. For example: You have just finish turned a workpiece with tool number five and the target value an external diameter is 2.0 plus or minus 0.002 inches. The diameter is measured to be 2.0033.What must be done? (and have students do what must be done).

To complete lesson 4.3, students must:

- view the on-line presentation for this lesson (18 minutes)
- read the reading materials for this lesson (20 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (25 minutes)
  - If using NC Guide, have students type the program with NC Guide and verify the program (20 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: 93 **minutes**

Notes:

Tool Nose Radius Compensation

Lesson Plan

4.4

Explains how the radius on the tool tip of single point turning tools (internal and external) affects the surfaces being machined and shows how to compensate.

Lessons in Key Concept 4 (you are here)

4: Know the compensation types
- 4.1: Introduction to compensation
- 4.2: Geometry offsets
- 4.3: Wear offsets
- **4.4: Tool nose radius compensation**

If supplementing the on-line content with lectures:

You may have described how the radius of a single point turning tool or boring bar affects machining during lesson 3.1 (the motion types). In this lesson, we'll be describing how tool nose radius compensation is used to compensate for the tool nose radius.

Explain that with any single point cutting tool, there is a small radius on the tip of the tool. And it is this tooling attribute that you'll be dealing with in this lesson.

Lesson objective:

To ensure that students understand why tool nose radius compensation is required as well as how it is used.

Main topics for this lesson:

- Why tool nose radius is required
- Steps to programming
- Setup person's considerations
- Programming offset entries

Key points made for each topic:

Why tool nose radius compensation is required

- We illustrate how the small radius on single point cutting tools is not always in contact with the workpiece when tool nose radius compensation is not used.
- We point out that this is only true with angular and circular surfaces.

Steps to programming

- We introduce the three simple steps needed to program tool nose radius compensation – instate, cut the work surface, and cancel.
- We explain that instating tool nose radius compensation involves including the appropriate instating word (G41 or G42) in the cutting tool's approach movement to the workpiece. When looking in the direction of the cutting tool's movement, G41 will be used if the cutting tool is on the left side of the work surface. G42 will be used if it is on the right. If machining occurs toward the chuck (as it normally does), G41 will be used for internal boring and G42 will be used for external turning.
- We demonstrate how to cut the work surface. This involves simply programming the work surface path with linear and circular motion commands. We also provide some warnings in the slide show about common mistakes made when programming the work surface path.
- We show that canceling tool nose radius compensation involves including a G40 word in the cutting tool's retract motion to the turret index position.

Programming example

- We illustrate the programming of tool nose radius compensation with a nice example.

Tool nose radius compensation from the setup person's point of view

- We point out that the R and T registers of the cutting tool's offset (we recommend using the geometry offset) must be correctly set in order for tool nose radius compensation to work.
- We explain that it is the setup person's responsibility to enter these offset values.
- We explain that the R register must contain the tool nose radius.
- We explain that the T register must contain a code number that tells the machine the style of cutting tool being used (turning tool, boring bar, etc.).
- There is a chart showing the code numbers in the reading materials as well as in the presentation. We recommend that you have students try to remember the two most common ones: 2 for a boring bar and 3 for a turning tool.

Programming tool nose radius compensation offset entries

- We point out that when a programmer chooses the style of cutting tool to be used for a given application, the radius of the cutting tool will remain fixed. That is, when an operator replaces a dull tool, the new tool will have the same radius as the dull tool. This, of course, means the programmer knows the tool nose radius as the program is written.
- We demonstrate the use of the G10 command – a command that allows the programming of offset entries.
- We point out that programming tool nose radius compensation offset entries will keep the setup person from having to enter them – saving time and minimizing the potential for mistakes.

Lesson Plan 4.4 (continued)

At the machine (about 10 minutes)

Run a program that uses tool nose radius compensation to machine angular and circular surfaces in single block mode. Using single block, after a motion that machines a taper or radius, toggle between the program page (that shows the programmed coordinate) and the position page (that shows the tool's actual position). Point out that the difference between these coordinates is related to tool nose radius compensation.

- The machine is automatically recalculating programmed coordinates in order to keep the cutting tool's radius tangent to each surface being machine.

Lab exercise

We have no suggestions for a lab exercise related to the material in this lesson.

To complete lesson 4.4, students must:	Exercises
▪ view the on-line presentation for this lesson (12 minutes) ▪ read the reading materials for this lesson (15 minutes) ▪ Do the exercise at the end of the reading materials for this lesson (5 minutes)	▪ Have students take the on-line test for this lesson. (10 minutes) ▪ Have students complete the programming activity for this lesson (25 minutes) - If using NC Guide, have students type the program with NC Guide and verify the program (20 minutes) - Have students submit the completed (typed) program for grading Approximate total study time for this lesson: **87 minutes**

Notes:

Key Concept
5

Lesson Plan: You Must Provide Structure to Your CNC Programs

Introduces Key Concept number five.

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well. As you can see, there are ten Key Concepts further divided into 28 lessons.

Key Concept number five is two-lesson key concept.

Points made during the introduction to Key Concept number five

- To this point in the class, we have been presenting the building blocks needed to write CNC programs. Students have worked on several programs, filling in the blanks for important words and commands – but they have not written a program from scratch.
- In Key Concept number five, we're going to show them how to become self-sufficient CNC programmers – able to write programs on their own.
- Since this Key Concept requires an understanding of everything we've presented so far, now is a great time to do a lengthy review – make sure your students are truly ready for Key Concept number five.
- Name a few programming words: G00, G01, G02, G03, M03, the O word, F word, S word – and so on. See if students can describe them. It's likely that they can. Point out that they already know quite a bit about the structure related to writing CNC programs.
- Remind students that learning to write CNC programs is like learning a foreign language that has only about 50-60 words. They've already learned the majority of these words.

Lessons for this Key Concept

5: You must provide structure to your CNC programs
- 5.1: Introduction to program structure
- 5.2: Structured program format

Key Concept five objective: Help students master the ability to write programs by themselves.

Notes:

Introduction to Program Structure

Lesson Plan

5.1

Explains why programs must be formatted using a strict and consistent structure.

Lessons in Key Concept 5 (you are here)

5: You must provide structure to your CNC programs

- **5.1: Introduction to program structure**
- 5.2: Structured program format

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Point out that only the structure by which programs are formatted dictates how safe, easy to use, and efficient programs will be. Unfortunately, what is done to enhance one objective commonly detracts from the others. Our emphasis will be on safety and ease of use. So our recommended formats will not be as efficient as possible.

- We'll be reviewing certain format related topics as well as presenting a few new ones in this lesson.

Lesson objective:

To ensure that students understand why programs must be strictly formatted to achieve the desired objectives.

Main topics for this lesson:

- Importance of strictly formatting programs
- Reasons for formatting
  - Getting familiar with programming
  - Consistency
  - Rerunning tools
- Machine differences

Key points made for each topic:

Importance of formatting

- We begin by presenting the three reasons why CNC programs must be strictly formatted: To allow familiarization with programming, to ensure consistency among programs, and most importantly, to allow cutting tools to be run by themselves (actually, to be re-run).
- For familiarization, we present a simple analogy related to driving an automobile. It is unlikely that any driver can recite from memory all of the road signs – but when they see a road sign, they quickly recognize its meaning. In similar fashion, few CNC programmer can recite from memory all CNC words used in programming. But when they see the word – especially when it is in the correct context of a CNC program, they will easily recognize its use.
- For consistency, we point out that programmers must be able to repeat past successes. If a given format works properly, achieving all required objectives, using its format in future programs will ensure continued success. Also, setup people and operators (and anyone else working with programs) will quickly become familiar with programs if they are consistently formatted.
- For re-running tools, we provide several examples of why strict formatting is important. We point out that each tool in the program should be treated as a mini-program. All words and commands necessary to get the machine running (the same words and commands as for the first tool) must be programmed at the beginning of each tool – making the tool independent from the rest of the program. Sometimes this means programming seemingly redundant words and commands.

Four types of format

- We introduce the four types of program format that will be presented in lesson fifteen: Program startup format, tool ending format, tool startup format, and tool ending format. We explain how these formats can be used as a crutch until they are memorized.

Machine differences

- We point out some of the machine differences that require different program formatting – like automatic tool changer differences and differences in M code numbering.

Efficiency improvements

- We stress that, as stated, our formats for programming emphasize safety and ease of use. This section presents some efficiency related limitations of our given formats – and provides suggestions for improvement.

Lesson Plan 5.1 (continued)

At the machine (about 20 minutes)

Pick one of the practice programs students have done in class and load it into the machine. Point out the strict structure used. Show students the restart command for each tool. Run the program once to show students the motions made by each tool.

Using a written step-by-step procedure for re-running tools, demonstrate the task of re-running the tools in the program.

Have students take turns practicing rerunning tools (again, using your written procedure to do so.

Lab exercise

Have students begin practicing program running procedures. Again, you must cautiously monitor their work.

Suggested procedures for hands-on practice:

- To run the program (in normal production - no verification techniques)

To complete lesson 5.1, students must:

- view the on-line presentation for this lesson (10 minutes)
- read the reading materials for this lesson (15 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (25 minutes)
  - If using NC Guide, have students type the program with NC Guide and verify the program (20 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: **80 minutes**

Notes:

Structured Program Format

Lesson Plan

5.2

Introduces the four types of program format and show students how to use them.

Lessons in Key Concept 5 (you are here)

5: You must provide structure to your CNC programs

- 5.1: Introduction to program structure
- **5.2: Structured program format**

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Students have worked on several programs – but they haven't written one on their own yet. They should know the majority of CNC words and commands used in programs, so you shouldn't have too many new ones to describe in this lesson. Concentrate on providing students with the formats – crutches they can use until they have the formats memorized.

- Point out that students should only be on their own to develop the commands that perform the machining operations in the program. The balance of any program is related to format.

Lesson objective:

To help students become self-sufficient programmers.

Main topics for this lesson:

- Initialized states
- Documentation
- Four kinds of format
  - Program startup
  - Tool ending
  - Tool startup
  - Program ending
- Formats for turning centers that have geometry offsets

Key points made for each topic:

A few more notes

- We make a few more points about program structure before showing the actual formats.
- If students have been doing the exercises and programming activities, it's likely that they have already learned some of these points.

Program formats

- We provide four sets of format for a turning center that uses geometry offsets with which to assign program zero.
- We point out that certain values (numbers) within CNC words will change from tool to tool and program to program. But the structure will remain the same. The presentation nicely illustrate this with color coding. In the reading materials, changing values are provided in bold fonts.
- As we come across any new word, of course, we explain it (like M01 and M30). This should finalize any concerns and questions that students have about the most common CNC words.
- We stress that with an understanding of these formats, writing a program will be a (relatively) simple matter of beginning with the program startup format, followed by the tool startup format. The student will be on their own to develop the commands to machine with the first tool. Then they follow the format to end the tool (tool ending format). Next they follow the format to start the next tool (tool startup format). They're on their own again to develop the commands to machine with the second tool. They'll toggle among tool ending, tool startup, and cutting commands until they're finished with the program – at which time they'll follow the tool ending format followed by the program ending format. Again, they are only on their own for the cutting commands within each tool.

Example program: We use the example program to make these points:

- Certain seemingly redundant words are required from tool to tool. If, for example two tools run in sequence that require the same spindle speed or feedrate, the related words (S and F) must be specified in both tools.
- Much of a typical program is simply format that can be copied from an existing program. (There are only six cutting commands in the example program – the rest of the program is related to format.).

At the machine (about 20 minutes)

Students will be writing their first program entirely on their own in the programming activity for this lesson. This will make a great program to work with at the machine.

Warning: To this point, programs that you have worked with on the machine have been previously verified (having no problems). This will be a student's program, and as such, may include mistakes. If your facility has some form of tool path verification system, like NC Guide or as part of the CNC machine, be sure to use it prior to actually running the program on the machine. Even then, be ready for anything when you do run the program.

Actually, it can be helpful if the program does include some mistakes. While you're not yet describing program verification (that's coming later, in Key Concept number ten), it doesn't hurt to allow students to see what they'll be up against when running their own programs.

After the program has been verified – and using a step-by-step procedure for running programs, have students practice running the program. And using the procedure to re-run tools, have them practice running one tool at a time (using optional stop to stop the machine after each tool.

Lab exercise

We have no suggestions for lab exercises that are related to this lesson. If you wish, you can have students continue practicing with machine operation procedures.

To complete lesson 5.2, students must:

- view the on-line presentation for this lesson (11 minutes)
- read the reading materials for this lesson (15 minutes)
- do the programming activity at the end of the reading materials for this lesson (20 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (25 minutes)
  - This will be the first program they write completely on their own.
  - If using NC Guide, have students type the program with NC Guide and verify the program (20 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: **101 minutes**

Notes:

Key Concept

6

Lesson Plan: Special Features that Help with Programming

Introduces Key Concept number six.

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine configurations
- 1.2: Turning center speeds and feeds
- 1.3: CNC job work flow
- 1.4: Visualizing the execution of a program
- 1.5: Understanding the workpiece coordinate system
- 1.6: Determining geometry offsets
- 1.7: Setting geometry offsets
- 1.8: Introduction to programming words

2: You Must Prepare to Write Programs
- 2.1: Preparation for programming

3: Understand the motion types
- 3.1: Programming the three basic motion types

4: Know the compensation types
- 4.1: Introduction to compensation
- 4.2: Geometry offsets
- 4.3: Wear offsets
- 4.4: Tool nose radius compensation

5: You must provide structure to your CNC programs
- 5.1: Introduction to program structure
- 5.2: Structured program format

6: Special features that help with programming
- 6.1: One-pass canned cycles
- 6.2: Rough and finish turning and boring (G71 and G70)
- 6.3: Other multiple repetitive cycles (G72-G75)
- 6.4: Threading multiple repetitive cycle (G76)
- 6.5: Sub-programming techniques
- 6.6: Control model differences
- 6.7: Other special programming features

7: Know your machine from a setup person or operator's viewpoint
- 7.1: Tasks related to setup and running production
- 7.2: Buttons and switches on the operation panels

8: Know the three basic modes of operation
- 8.1: The three modes of operation

9: Understand the importance of procedures
- 9.1: The key operation procedures

10: You must know how to safely verify programs

10.1: Program verification

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well. As you can see, there are ten Key Concepts further divided into 28 lessons.

Key Concept number six is four-lesson key concept.

Points made during the introduction to Key Concept number six

- At this point, students should be able to write programs on their own. But they have just the rudimentary tools to do so. Point out that writing programs with only the tools you have seen so far will be quite tedious (I like to point out, for example, that rough turning a workpiece that requires six roughing passes will require at least 24 commands).
- In Key Concept number six, you'll be showing several features that shorten programs, make programming easier, and in general facilitate the programming process.
- A good review of all material presented so far may be in order. Confirm that students are truly ready to learn about some rather advanced CNC programming features.

Lessons for this Key Concept

6: Special features that help with programming
- 6.1: One-pass canned cycles
- 6.2: Rough and finish turning and boring (G71 and G70)
- 6.3: Other multiple repetitive cycles (G72-G75)
- 6.4: Threading multiple repetitive cycle (G76)
- 6.5: Sub-programming techniques
- 6.6: Control model differences
- 6.7: Other special programming features

Key Concept six objective: Help students master those programming features that are appropriate to their needs.

Notes:

One-Pass Canned Cycles

Lesson Plan

6.1

Helps students master the use and limitations of these cycles cycles.

Lessons in Key Concept 6 (you are here)

6: Special features that help with programming

- **6.1: One-pass canned cycles**
- 6.2: Rough and finish turning and boring (G71 and G70)
- 6.3: Other multiple repetitive cycles (G72-G75)
- 6.4: Threading multiple repetitive cycle (G76)
- 6.5: Sub-programming techniques
- 6.6: Control model differences
- 6.7: Other special programming features

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Student know how to program machining operations using G00, G01, G02, and G03. Now you'll begin presenting how the programming of certain machining operations can be simplified. However, the one-pass canned cycles are not so helpful. We include them for the sake of completeness, but there are much more helpful cycles coming up in future lessons.

- As the name implies, each one-pass canned cycle will cause the machine to make one pass – a turning pass, a facing pass, or a threading pass.
- Students may be wondering why you waited so long to show canned cycles. Don't let them minimize the importance of what they've learned so far.

Lesson objective:

Help students master the programming of one-pass canned cycles – and to know their limitations.

Main topics for this lesson:

- Cycle consistencies
- One-pass canned cycles

Key points made for each topic:

Cycle consistencies

- We point out that these consistencies apply to one-pass canned cycles as well as multiple repetitive cycles.
- We explain that all cycles can be used for internal work as well as external work.
- We point out that all canned cycles require the programmer to send the tool to a convenient starting position. This position will vary for each canned cycle, but it is from this position that the cycle will begin.
- We also make it clear that all canned cycles will leave the tool back at the convenient starting position when the cycle ends.

One pass turning and boring cycle

- We explain the G90 – one pass turning and boring cycle. An example is also provided in the student manual.

One pass facing cycle

- In similar fashion, we explain the G94 one pass facing cycle.

One pass threading cycle

- And finally, we explain the one-pass threading cycle, G92.
- Once again, we point out that all three of these cycles are quite limited – and there are multiple repetitive cycles that are much more helpful.

Lesson Plan 6.1 (continued)

At the machine (about 20 minutes)

You may want to demonstrate the use of each of these cycles with an example program at the machine. But again, we don't recommend spending too much time on them – the multiple repetitive cycles are much more helpful.

Lab exercise

We have no suggestions related to the material covered in this lesson.

To complete lesson 6.1, students must:

- view the on-line presentation for this lesson (4 minutes)
- read the reading materials for this lesson (10 minutes)

Exercises

- Have students take the on-line test for this lesson. (5 minutes)
- Have students complete the programming activity for this lesson (15 minutes)
  - If using NC Guide, have students type the program with NC Guide and verify the program (15 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: **53 minutes**

Notes:

Rough and Finish Turning and Boring (G71/G70)

Lesson Plan

6.2

Help students understand how these very helpful cycles are used.

Lessons in Key Concept 6 (you are here)

6: Special features that help with programming

- 6.1: One-pass canned cycles
- **6.2: Rough and finish turning and boring (G71 and G70)**
- 6.3: Other multiple repetitive cycles (G72-G75)
- 6.4: Threading multiple repetitive cycle (G76)
- 6.5: Sub-programming techniques
- 6.6: Control model differences
- 6.7: Other special programming features

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

As stated in lesson 6.1, the one-pass canned cycles have pretty much been replaced with multiple repetitive cycles. In this lesson, you'll present two very popular multiple repetitive cycles. G71 is used for rough turning and rough boring. G70 is used for finishing.

Students have worked on several programs that perform rough and finish operations. But they've used G00 and G01 to do so. Now you'll help them master the programming of these operations using special cycles

- With G71, the programmer will define the finish pass and provide one simple command that specifies how roughing must be done.
- The machine will determine how to rough turn or rough bore, regardless of how many passes is required

Lesson objective:

To ensure that students master the use of these two important multiple repetitive cycles.

Main topics for this lesson:

- Introduction to multiple repetitive cycles
- G71 rough turning or boring
  - How it works
  - Words involved
- G70 finishing cycle
- G71 for rough boring
- Examples
- Using tool nose radius compensation

Key points made for each topic:

Introduction to multiple repetitive cycles

- We introduce the various multiple repetitive cycles and specifies which ones will be very helpful.

How G71 works

- We graphically illustrate the two phases of G71. In phase one, the machine will take a series of roughing passes, leaving steps on the workpiece. In phase two, the machine will take a semi-finish pass, leaving the specified amount of finishing stock.

Words in the G71 command

- We introduce the meaning of each word needed in the G71 command, including P and Q (used to specify the starting and ending block for the finish pass definition), D (used to specify depth of cut), U and W (used to specify finishing stock), and F (the feedrate used for the entire roughing cycle).

The finish pass definition

- We point out that right after the G71 command, the programmer specifies the finish pass.

Finishing cycle – G70

- We point out that to keep from having to repeat the finish pass definition for the finishing tool, G70 lets the programmer specify the starting and ending block for the finish pass.

Using G71 and G70 for rough and finish boring

- We illustrate the small differences for using these multiple repetitive cycles for internal work.

What about tool nose radius compensation?

- We illustrate that it is still possible (and easy) to use tool nose radius compensation for the finishing tool/s when using G71 and G70.

Lesson Plan 6.2 (continued)

At the machine

Students will be writing two practice programs that incorporate G71 and G70. Load one into the machine and demonstrate its use. Be sure to modify the depth of cut to show how easy it is to generate a greater or smaller number of roughing passes by simply changing one word.

Lab exercise

Since the program written during this lesson contain several tools – and since multiple repetitive cycles make the program so much shorter – you should be able to stress how to re-run tools. Have students practice finding the restart block for each tool.

You may also want to throw in some "what if" scenarios. For example, what if a finish boring bar is machining a hole too deep (in Z) by 0.004 inch. Assuming the program is correct, what must be done?

Suggested procedures for hands-on practice:

- To rerun tools

To complete lesson 6.2, students must:

- view the on-line presentation for this lesson (17 minutes)
- read the reading materials for this lesson (15 minutes)
- complete the programming activity at the end of the reading materials (15 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (20 minutes)
  - If using NC Guide, have students type the program with NC Guide and verify the program (15 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: **77 minutes**

Notes:

Other Multiple Repetitive Cycles (G72-G75)

Lesson Plan

6.3

Help students understand the rough facing multiple repetitive cycle, among others.

Lessons in Key Concept 6 (you are here)

6: Special features that help with programming
- 6.1: One-pass canned cycles
- 6.2: Rough and finish turning and boring (G71 and G70)
- **6.3: Other multiple repetitive cycles (G72-G75)**
- 6.4: Threading multiple repetitive cycle (G76)
- 6.5: Sub-programming techniques
- 6.6: Control model differences
- 6.7: Other special programming features

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Students now know how to easily command rough turning and rough boring with G71. In this lesson you'll be presenting G72, which can be used in a similar fashion for rough facing (still followed by G70 for finish facing). The other three multiple repetitive cycles (G73, G74, and G75) are not so helpful. In this lesson, you'll be explaining why.

There are times when a workpiece to be rough machined is in the form of a large diameter yet short workpiece (like a flange). For this kind of workpiece, it makes more sense to rough face it than to rough turn it – which is the function of G72.

- G73 is used for "pattern repeating", but there are not many feasible applications for its use.
- G74 is used for peck drilling, but it only breaks the chip as the hole is drilled. It does not clear the chips from the hole.
- And G75 is called the grooving cycle, but it simply plunges the groove once. It will not form chamfers or radii on the groove's corners.

Lesson objective:

To ensure that students master four other multiple repetitive cycles - the most important of which is the G72 rough facing cycle.

Main topics for this lesson:

- G72 rough facing
- G73 pattern repeating
- G74 peck drilling
- G75 grooving

Key points made for each topic:

G72 rough facing cycle

- We explain the minimal differences between G71 and G72. An example program is also provided in the slide show and in the student manual.

G73 pattern repeating cycle

- While we show a pretty good application for G73 in the presentation and reading materials, frankly speaking, few programmers use G73. Some try to use it for castings and forgings with limited success.

G74 peck drilling cycle

- While G74 works nicely with gummy materials to break chips, it will not clear chips between pecks – as is necessary for deep holes. Note that we provide an example of deep hole peck drilling using G00 and G01.

G75 grooving cycle

- This cycle only makes one grooving pass – assuming that the grooving tool is the same width as the groove. Also, it will not make chamfers or radii on the groove's corners. For these reasons, most programmers program grooving operations long-hand – with G00 and G01.

Lesson Plan 6.3 (continued)

At the machine

Students will be writing a practice program that incorporates G72 and G70. Load it into the machine and demonstrate its use. As with your demonstration of G71, be sure to modify the depth of cut to show how easy it is to generate a greater or smaller number of roughing passes by simply changing one word.

Lab exercise

The programming activity of this lesson involves writing a program that uses G72 rough facing. If you have students do this exercise, have them run the program on one of your lab machines.

Suggested procedures for hands-on practice:

- To do a free flowing dry run (if you have them run the program written during the programming activity)

To complete lesson 6.3, students must:

- view the on-line presentation for this lesson (8 minutes)
- read the reading materials for this lesson (15 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (20 minutes)
  - If using NC Guide, have students type the program with NC Guide and verify the program (15 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: **68 minutes**

Notes:

Threading Multiple Repetitive Cycle (G76)

Lesson Plan

6.4

Help students understand how to easily program threading operations.

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Explain that G76 makes it easy to program threading operations, regardless of how many passes are needed to chase the thread.

- The threading tool is first positioned to the convenient starting position.
- The G76 command is then given, specifying the ending diameter (major or minor diameter of the thread), the ending point of the thread (Z), the thread depth (K), the depth of the first pass (D), the tool angle (A), and the pitch (F).

Lesson objective:

To ensure that students understand and can program threading operations using the G76 command.

Main topics for this lesson:

- How threading is done on a lathe
- G32: great for tapping
- G76 threading command
- Taper threading
- Multiple start threading

Key points made for each topic:

How threading is done on a lathe

- We begin by introducing threading – which some of your students may be quite familiar with.

G32 – great for tapping

- We explain that G32 is like G01 in that one motion will be made per G32. But unlike G01, G32 will disable feedrate override and feed hold, making it quite useful for programming tapping commands.

Point on the threading tool that is programmed

- We point out that students should program from the leading edge of the threading tool, not the tip (point).

G76 threading command

- This is the heart of the presentation for this lesson. We explain the words involved with G76, as well as how to program threads.
- We explain the programming of taper threads – as well as multiple start threads.

At the machine

Students will be writing a practice program that incorporates G76. Load it into the machine and demonstrate its use. As with your demonstration of G71, be sure to modify the depth of the first pass to show how easy it is to generate a greater or smaller number of passes by simply changing one word.

Lab exercise

We have no suggestions for lab exercises that are related to this lesson.

To complete lesson 6.4, students must:

- view the on-line presentation for this lesson (15 minutes)
- read the reading materials for this lesson (15 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (20 minutes)
  - If using NC Guide, have students type the program with NC Guide and verify the program (20 minutes)
  - Have students submit the completed (typed) program for grading

Approximate total study time for this lesson: **80 minutes**

Notes:

Sub-Programming Techniques

Lesson Plan 6.5

Presents the applications and use of subprograms

Lessons in Key Concept 6 (you are here)

6: Special features that help with programming
- 6.1: One-pass canned cycles
- 6.2: Rough and finish turning and boring (G71 and G70)
- 6.3: Other multiple repetitive cycles (G72-G75)
- 6.4: Threading multiple repetitive cycle (G76)
- **6.5: Sub-programming techniques**
- 6.6: Control model differences
- 6.7: Other special programming features

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Point out that there are times when a programmer must repeat a series of commands within a program. We've shown one time to this point in the class: multiple identical grooves. Point out that any time commands must be repeated, it may be a good application for using subprograms.

- With subprograms, the programmer can cause the machine to exit the main program (temporarily) to execute a subprogram.
- When the machine finishes executing the subprogram, it will return to the main program to the command after the calling command and continue.
- Subprograms are named and loaded in the same way as main programs (every program shown to this point is a main program).

Lesson objective:

To help students recognize, understand, and master the applications for subprograms.

Main topics for this lesson:

- Applications
- Related words
- Examples
  - Multiple identical grooves
- Introduction to parametric programming

Key points made for each topic:

Applications

- We start by describing the four application categories for subprograms – repeated commands and operations, control programs, and utility programs. The slide show helps you do so.
- We show the application for machining multiple identical grooves – nicely illustrating how helpful a subprogram can be. We also show how subprograms can help with flip jobs and with bar-feeding.
- We then show the application categories, but only lists example applications in each. Later in the presentation, we show complete examples.

Related words

- We introduce the four words used with subprograms (M98, M99, P, and L).
- Next, we present a simple example to help students understand the points you've made so far.
- Since using subprograms requires multiple programs to be loaded, we provide some suggestions for naming subprograms that will help avoid confusion.
- We then show the *nesting* limitation for subprograms (calling one subprogram from another).

Examples

- We present full examples in all application categories.
- One of the categories (repeated machining operations) requires that you introduce how to program in the incremental positioning mode (using U and W instead of X and Z).
- The *control programs* application example is related to turning centers that machine flip jobs (machining half then turning the part around in the middle of the cycles.
- We also show a subprogram application that is helpful with turning centers that have bar feeders. But of course, not all turning centers have this accessory.

Introduction to parametric programming

- We point out that in order to use a subprogram, *all* of the commands in the subprogram must be totally redundant. If anything changes from one time the subprogram is needed to the next, a subprogram cannot be used.
- We explain that there is an optional feature called parametric programming (FANUC's version of parametric programming is custom macro B). While parametric programming is beyond the scope of this class, students should at least be familiar with applications for parametric programming. We present the five application categories.

Lesson Plan 6.5 (continued)

At the machine (about 10 minutes)

In the exercises for this lesson, students will be writing programs that use subprograms. You can easily use them (or exercises of your own design) for practice at the machine. When executing the program, monitor the PROGRAM display screen page. Be sure to point out that when a subprogram is executed, the main program appears to disappear – and only the subprogram is shown on the display.

Lab exercise

We have no suggestions for lab exercises that pertain to the subject matter for this lesson.

To complete lesson 6.5, students must:

- view the on-line presentation for this lesson (8 minutes)
- read the reading materials for this lesson (20minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)
- Have students complete the programming activity for this lesson (20 minutes)
  - If using NC Guide, have students type the programs with NC Guide and verify the them (20 minutes)
  - Have students submit the completed (typed) programs for grading

Approximate total study time for this lesson: **78 minutes**

Notes:

Control Model Differences

Lesson Plan

6.6

Present the minor programming differences among FANUC control models.

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Point out that there are some minor differences with regard to how certain FANUC controls are programmed. Examples shown so far have been for control models that use one-line multiple repetitive cycles. Information in this lesson pertains to those control models that require two-line multiple repetitive cycles. You may, of course, skip this lesson if your students won't be working with control models that use one-line multiple repetitive cycles.

- You may want to expand this presentation to include other controls that you know your students will be working with. While it is beyond the scope of this curriculum, you may, for example, want to discuss some of the programming differences between FANUC controls and others (like Okuma and others).

Lesson objective:

To ensure that students understand the minor programming differences among FANUC controls

Main topics for this lesson:

- Two programming methods for multiple repetitive cycles
- Programming methods for sub-program programming

Key points made for each topic:

Two styles of programming

- We explain the second style of programming multiple repetitive cycles which commonly require two commands.

G Multiple repetitive cycle differences

- We show the actual differences for each multiple repetitive cycle.
- We point out that the second style of programming actually has some advantages – especially for G76 threading (allowing minimum depth of cut, final pass depth, and number of spring passes to be programmed).

Sub-programming differences

- We explain the minor format difference for the M98 command's P word and L word.

Lesson Plan 6.6 (continued)

At the machine

We have no suggestions related to the material presented in this lesson.

Lab exercise

We have no suggestions for lab exercises that pertain to the subject matter for this lesson.

To complete lesson 6.6, students must:	**Exercises**
▪ view the on-line presentation for this lesson (6 minutes) ▪ read the reading materials for this lesson (10minutes)	▪ Have students take the on-line test for this lesson. (10 minutes) Approximate total study time for this lesson: **26 minutes**

Notes:

Special Programming Features

Lesson Plan

6.7

Show a few more programming features that can facilitate programming.

Lessons in Key Concept 6 (you are here)

6: Special features that help with programming
- 6.1: One-pass canned cycles
- 6.2: Rough and finish turning and boring (G71 and G70)
- 6.3: Other multiple repetitive cycles (G72-G75)
- 6.4: Threading multiple repetitive cycle (G76)
- 6.5: Sub-programming techniques
- 6.6: Control model differences
- **6.7: Other special programming features**

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

In this lesson, we'll be showing some other helpful programming features – but you should point out right away that some of these features may not be of immediate use. In fact, some may be very helpful to one programmer in one company but never needed by another in a different company. But before your students can begin to apply a feature, of course, they must know it exists.

At this point, we've presented the most popular programming features. Indeed, most of what we have presented will be of immediate need to the vast majority of CNC programmers. In this lesson, we'll be presenting some more important features, but some of them will not be of immediate need to everyone. For example:

- Block delete is a feature needed and used by most companies.
- Statement labels are rarely needed.

You can, of course, skip – or minimize presentations for– topics you know students won't need.

Lesson objective:

To help students recognize and master those special programming features that will be important to them.

.

Main topics for this lesson:

- Optional block skip techniques
- Sequence number techniques
- Other G codes of interest
- Other M codes of interest
- Introduction to parameters

Key points made for each topic:

Optional block skip (block delete)
- We begin by explaining how block delete works.
- We then show several applications.
- We feel one of the most important applications for block delete is with trial machining – and we describe how block delete can help.

N word techniques
- Though not often required, we show a technique that can be used when machining order (tool sequence) must be changed. This a rather advanced technique that uses sequence numbers as *statement labels* – and allow a kind of unconditional branch (GOTO) in a CNC program.

Other G codes of interest
- There are several G codes that have not been presented to this point in the class. We point out right away that if the G code has not been yet introduced, it is probably not needed on a regular basis. At this point in the class, we introduce the lesser needed G codes.
- G04 (dwell): We recommend limiting applications for G04 to relieving tool pressure (some programmers are too quick to use G04 to program around machine problems – as is explained in the presentation.
- G09, G61: Though not often needed, we explain the *exact stop check function*.
- G10: This G code is introduced in Key Concepts one and four. We some advanced techniques for the *data setting command*.
- G20, G21: We've been mentioning the differences between inch and metric modes throughout the class. In this presentation, we recap – and we show the accuracy advantage of the metric mode.

Other M codes of interest
- As with G codes, there are a few more M codes that your students must be exposed to.
- M00: We present the application and use of the *program stop* command.
- M13, M14: Some, but not many, turning centers allow the spindle and coolant to be activated at the same time. We explain them.
- If you know that the machine/s your students will be using have M codes not addressed by this class, you should introduce them here.

Understanding parameters
- While parameters have little to do with programming, there are some that affect the way the machine behaves when executing programs. All CNC people should at least know what parameters are. Better, they should know the kinds of functions that parameters control. This presentation introduces parameters.

At the machine (about 10 minutes)

You may want to develop a special “demonstration program” that shows the use of features described in this lesson. But remember that some of these features are options, meaning you’ll have to limit your program to showing only those features that are equipped on your machine.

You may also want to demonstrate how find out whether a given FANUC control has an optional G code. In MDI, enter and execute the G code by itself. If you receive no alarm – or if the alarm is related to the format for the G code, the machine has it. If you receive alarm 10 (unusable G code), the machine does not have it.

Lab exercise

We have no recommendations related to the material presented in this lesson.

To complete lesson 6.7, students must:

- view the on-line presentation for this lesson (14 minutes)
- read the reading materials for this lesson (15 minutes)

Exercises (none)

- Have students take the on-line test for this lesson. (10 minutes)

Approximate total study time for this lesson: **39 minutes**

Appendix;

The Appendix immediately follows lesson 6.7 in the reading materials. Since it is related primarily to programming, we place it right after the programming presentations. Here is a list of the materials covered in the appendix. Based upon your knowledge of what your students will need upon completing this course, you can cover the related material.

Appendix: Special Machine Types And Accessories
Work holding and work support devices
  Work holding devices
  Three jaw chucks
  Collet chucks
  Work support devices
  Tailstocks
  The tailstock body
  Tailstock quill
  Center
  Tailstock alignment problems
  Programming considerations
  Steady rests
Bar feeders
  How a bar feeder works
  Workholding considerations
  Styles of bar feeders
  How to program for bar feeders
  Determining how much to feed the bar
  The steps to bar feeding
  The redundancy of bar feed programming
  When to program the bar feed
  Ending a bar feed program
  An example bar feeding program
  Ending a bar feed program
  An example bar feeding program
Part catchers
Live tooling
  Features of live tooling turning centers
  Rotating tools
  Special tool holders
  Precise control of main spindle rotation
  Only one way to specify speed and feedrate
  Hole machining canned cycles
  Polar coordinate interpolation
  Selecting the main spindle mode
  Programming an indexer
  Example program for an indexer
  Programming a rotary axis (C axis)
  Angular values
  Zero return position
  Rapid versus straight line motion
  Program zero assignment
  Absolute versus incremental
Canned cycles for hole machining
  How do you specify the machining direction?
  Canned cycle types
  Words used in canned cycles
  An example program
Understanding polar coordinate interpolation
Other machine types
  Twin spindle turning centers
  Sub-spindle turning centers
  Swiss-type turning centers

Notes:

Key Concept 7	**Lesson Plan:** ***Know Your Machine from an Operator's Viewpoint***
	Introduces Key Concept number seven.

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine configurations
- 1.2: Turning center speeds and feeds
- 1.3: CNC job work flow
- 1.4: Visualizing the execution of a program
- 1.5: Understanding the workpiece coordinate system
- 1.6: Determining geometry offsets
- 1.7: Setting geometry offsets
- 1.8: Introduction to programming words

2: You Must Prepare to Write Programs
- 2.1: Preparation for programming

3: Understand the motion types
- 3.1: Programming the three basic motion types

4: Know the compensation types
- 4.1: Introduction to compensation
- 4.2: Geometry offsets
- 4.3: Wear offsets
- 4.4: Tool nose radius compensation

5: You must provide structure to your CNC programs
- 5.1: Introduction to program structure
- 5.2: Structured program format

6: Special features that help with programming
- 6.1: One-pass canned cycles
- 6.2: Rough and finish turning and boring (G71 and G70)
- 6.3: Other multiple repetitive cycles (G72-G75)
- 6.4: Threading multiple repetitive cycle (G76)
- 6.5: Sub-programming techniques
- 6.6: Control model differences
- 6.7: Other special programming features

7: Know your machine from a setup person or operator's viewpoint
- 7.1: Tasks related to setup and running production
- 7.2: Buttons and switches on the operation panels

8: Know the three basic modes of operation
- 8.1: The three modes of operation

9: Understand the importance of procedures
- 9.1: The key operation procedures

10: You must know how to safely verify programs

10.1: Program verification

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well. As you can see, there are ten Key Concepts further divided into 28 lessons.

Key Concept number seven is two-lesson key concept.

Introduction to setup and operation

Key Concept number seven begins the setup and operation part of this course. However, you've done a great deal during the programming-related lessons to prepare students for setup and operation. Indeed, we've been giving suggestions in each lesson plan to help you stress setup and operation related topics.

We've done so for three reasons. First, and as stated, programmers *must* know enough about setup and operation to direct setup people and operators. Truly, the more a programmer knows about setup and operation, the better and more efficient the programs they will write.

Consider, for example, the technique shown in lesson 6.7 that is related to trial machining using block delete. With a full understanding of what a setup person or operator must do in order to trial machine, a programmer can include commands right in the program that facilitate any trial machining application. Without this understanding, the setup person and operator must struggle through trial machining on their own.

Second, setup people and operators can truly benefit from having a working knowledge of certain programming features. When appropriate, we've provided suggestions in each lesson plan to help you explain certain programming functions to setup people and operators.

While you didn't go into programming details for setup people and operators, you explained enough to help them understand the setup- and operation-related implications of these programming features. During Key Concepts one and four, for example, you explained enough about program zero assignment, wear and geometry offsets, and tool nose radius compensation to help setup people and operators understand the reasons why certain things must be done at the machine.

Third, we've minimized the need for duplicating presentations. If you've followed our recommendations and presented the setup- and operation-related implications of certain programming features during programming, you won't have to repeat these presentations during the setup and operation part of this class – though reviewing key points never hurts.

Lessons for this Key Concept

7: Know your machine from a setup person or operator's viewpoint
- 7.1: Tasks related to setup and running production
- 7.2: Buttons and switches on the operation panels

Key Concept six objective: To help students understand a turning center from a setup person's or operator's viewpoint.

Notes:

Tasks Related to Setup and Running Production

Lesson Plan

7.1

Explains the difference between setup tasks and production-running tasks.

Lessons in Key Concept 7 (you are here)

7: Know your machine from a setup person or operator's viewpoint

- **7.1: Tasks related to setup and running production**
- 7.2: Buttons and switches on the operation panels

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

Determining the distinction between setup-related tasks and production running-related tasks is pretty simple. When the machine is down between production runs, it is in setup. It is, of course, the setup person the makes setups.

However, we look at *operation* two ways.

- First, there are certain things an operator must master to be confident with the machine. They must, for example, know all the buttons and switches, they must master certain operation procedures, and in general, they must be comfortable running the machine.
- Second, once a setup is completed, the operator must run workpieces. Any task related to completing a production run is the responsibility of the CNC operator.

Lesson objective:

To help students understand the tasks related to setting up and running a CNC turning center.

Main topics for this lesson:

- Operator responsibilities
- Setup versus production running tasks
- Tasks related to setup
- Tasks related completing production runs

Key points made for each topic:

Introduction to setup and operation

- We introduce the four Key Concepts of setup and operation

Operator responsibilities

- We explain that CNC-using companies vary with regard to what they expect of their CNC people. In this lesson, we're going to explain all of the tasks needed to setup and run a CNC turning center.
- We point out that most (especially product-producing) companies break up these tasks. Several people are involved. But in some companies (especially workpiece producing and tooling producing companies), one person may be expected to perform all of the tasks we show in this lesson.

Tasks related to setup

- These tasks, of course, get the machine ready to run production.
- We present these tasks in the approximate order that setups are actually made.
- We include verifying the program and running the first workpiece as part of setup. Until a part passes inspection, of course, the operator cannot start the production run.

Tasks related to completing a production run

- We explain each task required to complete a production run.
- First we show tasks that must be completed in every cycle (like workpiece load/unload, activating the cycle, and workpiece measurement).
- We then show tasks that don't take place in every cycle (sizing adjustments, preventive maintenance, replacing dull tools, etc.).

While it's not shown in the presentation, the reading materials provide an excellent "sample scenario" for how a job gets setup and run. Using the same tasks described in the lesson, it walks students through the running of a sample job from start to finish.

Lesson Plan 7.1 (continued)

At the machine (about 15 minutes)

Use one of the programs from the programming activities, possibly the one for canned cycles, to review the tasks related to setup and running production. While you don't have to perform every step, at least show students the results of every task (completed program zero assignment in geomertry offsets, filled-in offset table, workholding device setup, cutting tools in the turret, program in memory, etc.).

While it is unlikely that you have hundreds of workpieces to run, be sure students understand that in the real world, companies commonly run hundreds – possibly thousands – of workpieces during a production run. During this time, tools show signs of wear and may require sizing adjustments. Eventually they dull completely and must be replaced. And if the production run lasts for days or weeks, it is likely that the machine will be turned off at some point. Many companies warm up their machines prior to starting a shift. It may be difficult, if not impossible, to illustrate all of this in class, but at least prepare students for what they'll face when they work for a CNC-using company.

Lab exercise

Have students practice with pre-developed procedures for basic tasks like power-up, jogging the axes, using the handwheel, starting the spindle, and so on.

To complete lesson 7.1, students must:

- view the on-line presentation for this lesson (10 minutes)
- read the reading materials for this lesson (30 minutes)
- do the exercise at the end of the reading materials (10 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)

Approximate total study time for this lesson: **60 minutes**

Notes:

Buttons and Switches on the Operation Panels

Lesson Plan 7.2

Explains all of the buttons and switches on a typical CNC turning center

Lessons in Key Concept 7 (you are here)

7: Know your machine from a setup person or operator's viewpoint
- 7.1: Tasks related to setup and running production
- **7.2: Buttons and switches on the operation panels**

If supplementing the on-line content with lectures:

Solicit questions about previous topics. Review the tasks related to setup and running production.

We've discussed many important buttons and switches during programming-related lessons. If you've been following our suggestions for things to do at the machine and having students do the lab exercises, it's likely that students have understand many of them. Now you'll be explaining *all* buttons and switches on the machine. If we miss some in the presentation and reading materials, be sure to cover them in class.

Setup people and operators must understand the function of *all* buttons and switches on the machine. Even if a particular button or switch is never used, they should understand *why* it is never used.

- A setup person or operator should never give up until they know the function of all buttons and switches.
- When they start running actual machines in the shop, there will likely be unfamiliar buttons and switches. They'll be on their own to learn about them (from the control manufacturer's manual and the machine tool builder's manual, or by talking to experienced people in the company.

Lesson objective:

To help students understand all of the buttons and switches on CNC turning centers – and master those that are most often-used.

Main topics for this lesson:

- The two operation panels
- Control panel buttons and switches
  - Keyboard
  - Display screen
- Machine panel buttons and switches
- Importance of the mode switch
- Three basic modes of operation
  - Manual mode
  - Manual data input (MDI) mode
  - Program operation mode

Key points made for each topic:

The two operation panels

- We break the operation panels on a CNC turning center into two categories – the control panel (made by the control manufacture – FANUC in our case), and the machine panel/s (made by the machine tool builder.
- There could be several machine panels – the main one close to the display screen, as well as others located as needed (like near the bar feeder if the machine has one).

Buttons and switches on the control panel

- We explain that these are buttons and switches located on the operation panel made by the control manufacturer, FANUC.
- We describe each button and switch on a typical control panel.
- You must, of course, explain the function of buttons and switches that are on you machine's control panel that are not explained in the slide show or student manual. Show students where this information can be found in the related manuals.

Buttons and switches on the machine panel

- We explain that these are buttons and switches located on the operation panel made by the machine tool builder.
- We describe each button and switch on a typical machine panel.
- We point out that machine panels vary dramatically from one machine tool builder to another. Builders can't seem to agree on what a CNC setup person or operator needs to run the machine.
- Be sure to explain the function of buttons and switches that are on the machine panel that are not explained in the presentation or reading materials. Show students where this information can be found in the related machine tool builder's manual/s.
- We limit the presentation for the mode switch. We feel that it is such an important switch, we devote an entire lesson to discussing it in Key Concept number eight.

Lesson Plan 7.2 (continued)

At the machine (about 30 minutes)

Indeed, this entire lesson could be presented (by you) at the machine. You can even have students to forego the on-line presentation of this lesson if you do. You can go over each operation panel, thoroughly describing each function. When you're finished, ask students if there are any buttons and switches they don't recognize.

We will also be describing every display screen page in this lesson. Most of these display screens have been discussed during the programming-related lessons.

It's unlikely that students will remember (memorize) every button and switch when you do this the first time. Be sure to review the buttons and switches as you begin upcoming lessons. Explain that they can use the on-line presentation to let them review on their own.

Lab exercise

If you have several different CNC machines, have students develop a few familiar procedures (like starting the machine and jogging an axis) for a machine with which they are not familiar. You must guide them of course, showing the procedure for them to document. Let some time pass and have them demonstrate the procedure/s to see how well they did.

Explain that students will eventually be running unfamiliar machines. The must be able to develop procedures for these tasks on their own.

To complete lesson 7.2, students must:

- view the on-line presentation for this lesson (20 minutes)
- read the reading materials for this lesson (15 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)

Approximate total study time for this lesson: 45 **minutes**

Notes:

Key Concept

8

Lesson Plan: Know the Three Basic Modes of Operation

Introduces Key Concept number eight.

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well.

Key Concept number eight is one-lesson key concept.

Points made during the introduction to Key Concept number eight

The most important switch on any turning center is the mode switch. This switch must be placed in the appropriate position before the desired function can be activated. While there are more than three positions on the mode switch, this switch can be divided into three basic categories: manual, manual data input (MDI), and program operation. It is the focus of this one-lesson Key Concept to describe these modes.

Point out that the mode switch is at the heart of any CNC turning center. No function can be activated unless this switch is in the appropriate position.

- The mode switch is always the first switch a setup person will set when performing any function on the machine.
- If the mode switch is improperly set, the machine will not respond to the desired action. (This is nice to know. The worst that can happen when the mode switch is not correctly set is that the machine won't respond – it simply won't do *anything*.)

Lessons for this Key Concept

8: Know the three basic modes of operation
- 8.1: The three modes of operation

Key Concept seven objective: To ensure that students have an understanding of the three basic modes of machine operation.

Notes:

The Three Modes of Operation

Lesson Plan

8.1

Explains every position on the mode switch.

Lessons in Key Concept 8 (you are here)

8: Know the three basic modes of operation

8.1: The three modes of operation

If supplementing the on-line content with lectures:

Solicit questions about previous topics. Review setup and operation tasks – and review the buttons and switches on the machine.

The most important switch on the machine is the mode switch. You'll be explaining it in detail during this lesson.

We begin by explaining the importance of the mode switch. Then we present the three most basic modes.

Lesson objective:

To ensure students have an understanding of the three basic modes of machine operation.

Main topics for this lesson:

- Importance of the mode switch
- Manual mode
- Manual data input (MDI) mode
- Program operation mode

Key points made for each topic:

The importance of the mode switch

- We begin by showing the two most popular types of mode switches – a rotary switch and a series of lighted buttons.
- We point out that if the mode switch is in the wrong position, the machine won't respond to an action.
- We explain that the mode switch is the first switch to be set when performing any function on the machine.

The three modes of operation

- **Manual mode:** We explain that manual mode, which includes (at least) jog, handwheel, and zero return, is used to get a quick response from the machine. In any of these modes, a button is pressed and the machine responds (by starting the spindle, moving an axis, turning on the coolant, etc.). We present several examples of when manual mode is used.
- **Manual data input (MDI) mode**: We explain that this mode includes the mode switch positions MDI and edit. We explain that the MDI mode switch position is used primarily to manually activate functions for which there are no manual controls. Most turning centers, for example, provide no manual means to activate the automatic tool changer. If an operator wants to cause a manual tool change they must use the MDI mode switch position to do so. We then show some examples of using this function. We explain that the edit mode is used to modify CNC programs. If you've been running any practice programs on the machine, it's likely that you've demonstrated this function by now. We provide a good example.
- **Program activation mode**: We explain that this mode is used to run programs. With current machines, there is only one mode switch position, labeled either auto or memory. Very old machines may have a tape mode.

Lesson Plan 8.1 (continued)

At the machine (about 10 minutes)

Using previously developed procedures to do so, demonstrate the various modes. If you've been following suggestions made in previous lessons, students have already practiced with some of these procedures.

Be sure to demonstrate what will happen (nothing) when the mode switch is in the wrong position.

Lab exercise

We have no suggestions for this lesson.

To complete lesson 8.1, students must:

- view the on-line presentation for this lesson (11 minutes)
- read the reading materials for this lesson (10 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)

Approximate total study time for this lesson: **33 minutes**

Notes:

Key Concept 9	***Lesson Plan:*** ***Understand the Importance of Procedures***
	Introduces Key Concept number nine.

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well.

Key Concept number nine is one-lesson key concept.

Points made during the introduction to Key Concept number nine

We cannot overstress the importance of procedures. We urge you to develop a set of step-by-step procedures for each machine you have in your lab.

In the reading materials, we show a set of procedures for the FANUC Certification Cart, made by Levil. You can it as a template to develop your own procedures. That is, develop procedures for each task presented (power-up, doing a zero return, jogging the axes, using the handwheel, starting the spindle, etc.).

Note that these procedures are listed in several categories (manual procedures, MDI procedures, setup procedures, program modification procedures, program running procedures, and program verification procedures). You may find the need to develop more procedures based upon the accessories equipped with your machine.

- Whenever you see a student struggling with a machine function, it should be taken as a signal that you need to develop a procedure to help.
- Eventually, setup people and operators will memorize the most often-needed procedures. But until then, they'll need some help.

I'm always amazed when I go into CNC using companies to find that they have not documented the procedures needed to run their CNC machines. While entry level people do eventually muddle through and memorize these procedures, they must put up with a great deal of frustration and hazardous situations. And the procedures they come up with may not be the best or most efficient methods of doing things.

You must set a good example. Provide students with procedures for *everything*. And make students learn how to document the procedures needed to run a machine on their own.

Lessons for this Key Concept

9: Understand the importance of procedures 9.1: The key operation procedures	**Key Concept seven objective:** To help students understand that having a procedure to perform any task will simplify the task – more importantly – that they will probably need to develop their own set of procedures when they start working for a CNC-using company.

Notes:

The Importance of Procedures

Lesson Plan

9.1

Explains every position on the mode switch.

Lessons in Key Concept 9 (you are here)

9: Understand the importance of procedures

9.1: The key operation procedures

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

In many companies, entry level people really struggle when leaning how to run a machine. How do you power up the machine? A person may have to be shown this procedure several times before they can remember how to do it. And this is but one (rather simple) procedure. This "show me" method of learning is frustrating for everyone involved.

In this lesson, we'll be showing students the most important procedures (procedures they should document for themselves for any machine they must run) as well as demonstrating their use.

Lesson objective:

To provide students with the procedures they need to run a CNC turning centers.

Main topics for this lesson:

- The importance of procedures
- Manual procedures:
- Manual data input procedures
- Setup procedures
- Program manipulation procedures
- Program running procedures

Key points made for each topic:

Importance of procedures

- We explain that with an understanding of what must be done (which we've been showing throughout the class), running a turning center is little more than following a series of procedures.
- We point out that step-by-step procedures will help newcomers perform any machine function – as long as they know why the function must be performed.
- We provide a "road map" analogy to help stress the importance of procedures.
- We divide the procedures into categories, beginning with manual procedures. We demonstrate every procedure in the presentation. We document them in the reading materials. It might be best to actually demonstrate procedures (at least those that you haven't shown to this point in the class) right on your lab machine.

Manual procedures

- These are procedures that will render an immediate response.

Manual data input (MDI) procedures

- These are procedures that use the MDI mode – usually required for functions that must be manually activated, but for which have no manual controls.

Setup procedures

- We've provided a few procedures needed during setup (like measuring program zero assignment values), but you may elect to develop more.

Program manipulation procedures

- These are procedures used to load, call up, and edit CNC programs.

Program running procedures

- Procedures needed to verify and run CNC programs are shown in lesson twenty-four.

Lesson Plan 9.1 (continued)

At the machine (about 10 minutes)

If you haven't already, be sure to demonstrate the use all procedures shown in this lesson.

Lab exercise

Have students develop a few procedures on their own. Make them document the step-by-step procedure for power-up, jogging the axes, using the handwheel, and other important procedures. You can provide them with a blank form and submit procedures for grading – or simply have them write down the procedures in a notebook.

To complete lesson 9.1, students must:	Exercises
▪ view the on-line presentation for this lesson (26 minutes) ▪ read the reading materials for this lesson (15 minutes)	▪ Have students take the on-line test for this lesson. (10 minutes) Approximate total study time for this lesson: **51 minutes**

Notes:

Key Concept

10

Lesson Plan: You Must Know How to Safely Verify Programs

Introduces Key Concept number ten.

In the course outline to the left, the Key Concepts are shown in bold. The lessons included in each Key Concept are shown as well.

Key Concept number ten is one-lesson key concept.

Points made during the introduction to Key Concept number nine

The final Key Concept draws together much of what has been presented in this class. Students must know how to safely verify new programs as well as programs that have been run before. They must also must, of course, be able to machine acceptable workpieces. Though companies vary in this regard, we're assuming that it is quite important to make the first workpiece being machined a good one.

Students must be able to find and correct mistakes. Mistakes can be related to the program or to the setup that has been made. This means they must be able to recognize the cause of problems being encountered – and again – this requires a good understanding of what has been presented so far.

We provide a series of procedures for verifying CNC programs (dry run, air cutting normal run, and cautiously running the first workpiece). These procedures are not overly specific – and are somewhat complex. And again, they require students to understand many of the points made so far.

Lessons for this Key Concept

10: You must know how to safely verify programs
- 10.1: Program verification

Key Concept seven objective: To help students understand how to safely verify new and previously run programs – and make the very first workpiece being machined a good one.

Notes:

Program Verification

Lesson Plan

10.1

Explains how to safely verify CNC programs.

Lessons in Key Concept 8 (you are here)

10: You must know how to safely verify programs

10.1: Program verification

If supplementing the on-line content with lectures:

Solicit questions about previous topics.

While it's not shown in the on-line presentation, the reading materials provides an realistic example of verifying a program from start to finish – including the running of the first workpiece. A similar example scenario is shown in the reading materials for lesson 7.1, but in that example, there are no mistakes. With the example job shown in this lesson, *there are many mistakes*. This should give students a realistic view of what they'll be in for when they begin working on a CNC turning center.

Lesson objective:

To provide students with the procedures they need to safely verify CNC programs.

Main topics for this lesson:

- Safety priorities
- Typical mistakes
- New versus proven programs
- Verifying new programs
- Verifying previously run programs
- Completing a production run

Key points made for each topic:

Safety priorities

- We begin by relating the three levels of priority a setup person should adhere to – operator safety, machine safety, and workpiece safety.

Typical mistakes

- We review the most common mistakes that are made when programming and making setups.
- We point out that when a mistake is found during program verification, the setup person will only see the "symptom" of the problem. Determining the problem requires a kind of backtracking from the symptom to potential causes. Knowing the typical mistakes that can be made helps limit the potential causes.

New versus proven programs

- We explain the differences. Generally speaking new programs are more difficult to verify - but even proven programs may present problems if mistakes are made during setup.

Review of program verification functions

- We introduce and/or review machine functions like the program check page, feed hold, single block dry run, feedrate override and rapid override.
- We recommend that when activating programs (by pressing the cycle start button), that students always keep a finger ready to press the feed hold button.

Program verification procedures

- Based upon whether the program being verified is a new or proven program, we show procedures to verify it.
- The reading materials include an excellent example scenario to illustrate the program verification procedure – including trial machining for critical machining operations.

Lesson Plan 10.1 (continued)

At the machine (about 30 minutes)

Use an example program that contains mistakes (possibly the one provided in the reading materials for this lesson - it contains mistakes) to demonstrate program verification – as well as how to correct mistakes.

If you have been running the programs students have written during class, it's likely that you've already done some of this.
Be sure to emphasize setup mistakes (like improper tool length compensation entries and program zero assignment mistakes.

Lab exercise

While you'll want to be very careful if the program contains mistakes, have students verify a program on their own.

Suggested procedures for hands-on practice:

- To do a free flowing dry run
- To do a normal air-cutting run
- To run the first workpiece
- To cancel a cycle

To clear an alarm

To complete lesson 10.1, students must:

- view the on-line presentation for this lesson (27 minutes)
- read the reading materials for this lesson (15 minutes)

Exercises

- Have students take the on-line test for this lesson. (10 minutes)

Approximate total study time for this lesson: **52 minutes**

Notes:

Summary of time required for students to complete lessons

Times are approximate and assume students have moderate aptitude for learning CNC and average reading skills.

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine configurations (47 minutes)
- 1.2: Turning center speeds and feeds (40 minutes)
- 1.3: CNC job work flow (32 minutes)
- 1.4: Visualizing the execution of a program (34 minutes)
- 1.5: Understanding the workpiece coordinate system (53 minutes)
- 1.6: Determining geometry offsets (66 minutes)
- 1.7: Setting geometry offsets (69 minutes)
- 1.8: Introduction to programming words (49 minutes)

2: You Must Prepare to Write Programs
- 2.1: Preparation for programming (97 minutes)

3: Understand the motion types
- 3.1: Programming the three basic motion types (108 minutes)

4: Know the compensation types
- 4.1: Introduction to compensation (79 minutes)
- 4.2: Geometry offsets (92 minutes)
- 4.3: Wear offsets (93 minutes)
- 4.4: Tool nose radius compensation (87 minutes)

5: You must provide structure to your CNC programs
- 5.1: Introduction to program structure (80 minutes)
- 5.2: Structured program format (101 minutes)

6: Special features that help with programming
- 6.1: One-pass canned cycles (53 minutes)
- 6.2: Rough and finish turning and boring (G71 and G70) (77 minutes)
- 6.3: Other multiple repetitive cycles (G72-G75) (68 minutes)
- 6.4: Threading multiple repetitive cycle (G76) (80 minutes)
- 6.5: Sub-programming techniques (78 minutes)
- 6.6: Control model differences (26 minutes)
- 6.7: Other special programming features (39 minutes)

7: Know your machine from a setup person or operator's viewpoint
- 7.1: Tasks related to setup and running production (60 minutes)
- 7.2: Buttons and switches on the operation panels (45 minutes)

8: Know the three basic modes of operation
- 8.1: The three modes of operation (33 minutes)

9: Understand the importance of procedures
- 9.1: The key operation procedures (51 minutes)

10: You must know how to safely verify programs

10.1: Program verification (52 minutes)

Total time to complete the course: Approximately 30 hours

Notes: