# Lesson Plans

## Machining 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
- 1.1a Levil machining center certification cart
- 1.2: CNC job work flow
- 1.3: Visualizing the execution of a program
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- 1.7: 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: Tool length compensation
- 4.3: Cutter radius compensation
- 4.4: Workpiece coordinate system offsets

**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: Hole-machining canned cycles
- 6.2: Working with subprograms
- 6.3: Other special programming features
- 6.4: Programming rotary devices

**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 24 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 machining 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 machining centers.

### Lessons for this Key Concept

**1: Know Your Machine from a Programmer's Viewpoint**
- 1.1: Machine configurations
- 1.1a Levil machining center certification cart
- 1.2: CNC job work flow
- 1.3: Visualizing the execution of a program
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- 1.7: 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.

**Key Concept 1 (continued)**

**Notes:**

# *Machine Configurations*

## Lesson Plan 1.1

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

Lessons in Key Concept 1 (you are here)

1: Know Your Machine from a Programmer's Viewpoint
- **1.1: Machine Configurations**
- 1.1a: Levil Machining Center Certification Cart
- 1.2: CNC job work flow
- 1.3: Visualizing program execution
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- 1.7: Introduction to programming words

**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 machining 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
  - C-frame style VMC
  - Gantry style VMC
  - Horizontal
- Directions of motion
  - C-frame VMC
  - Gantry style VMC
  - Horizontal
- Programmable functions
  - Spindle
  - Feedrate
  - Coolant
  - Automatic tool changer
  - 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 (hole machining operations like drilling, tapping, reaming, etc., as well as all forms of milling operations.
- Students 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 the main components by name.
- Students learn the difference between vertical and horizontal machining centers – as well as the differences among the various types of CNC machining centers.

**Directions of motion**
- Students learn the directions of motion (axes) for each kind of machining center.
- With many machines, the cutting tool does not move along with the axis (the table of a vertical machining center commonly moves in XY, while the cutting tool remains in a fixed position).
- Students learn the polarity (plus versus minus) of each axis – be sure students understand. And that, since the cutting tool does not move along with every axis, polarity can be a bit confusing.

**Programmable functions**
- We explain that CNC programmers must know the functions of their CNC machine/s that are programmable. Presentations in the slide show include the four most common programmable functions– spindle, feedrate, coolant, and tool changing. If your machines have more programmable functions (like automatic doors), be sure to include them in your own presentations.
  While this presentation includes an introductions to the related programming words, students need not try to memorize them.

**Spindle – students will be shown that…**
- Most machining centers allow the spindle to be programmed in three ways, speed (with S), activation (with M03, M04, and M05), and range (also with S).
- Speed is specified in revolutions per minute (rpm).
- M03 (forward) is used for right-hand tools and M04 (reverse) is used for left-hand tools – and that since right hand tools are much more popular than left-hand tools, M03 is more often used to activate the spindle.
- Range selection is rather transparent – part of the S word.

**Feedrate – students will be shown that…**
- An F-word is used to specify feedrate and feedrate is specified only in per-minute fashion (inches per minute or millimeters per minute) with most machining centers.

**Coolant – students will be shown that…**
- Coolant is used to cool and lubricate the machining operation. M08 turns on flood coolant and M09 turns it off.

**Tool changing – students will be shown that…**
- All true machining centers have automatic tool changers but they vary with regard to how they are programmed. Our slide shows help you present the two most popular types – single arm and double arm tool changers.

**At the machine (about 15-minutes)**

When students have completed 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. Pay particular attention to any axis with which the cutting tool does not move (polarity for these axes can be confusing to students).

Be sure to demonstrate 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 the automatic tool changer.

**Lab exercise (about 5-minutes per student)**

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 (17 minutes)
- read the reading materials for lesson 1.1 (20 minutes)
- take the quiz at the end of the reading materials (5 minutes)
- if applicable, read the reading materials for lesson 1.1a (the certification cart) (10 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: **62 minutes**

**Notes:**

## *CNC Job Work Flow*

**Lesson Plan**

**1.2**

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

Lessons in Key Concept 1 (you are here)

1: Know Your Machine from a Programmer's Viewpoint

- 1.1: Machine Configurations
- 1.1a: Levil Machining Center Certification Cart
- **1.2: CNC job work flow**
- 1.3: Visualizing program execution
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- 1.7: 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 machining 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
- Make workholding setup
- Assign program zero
- Assemble cutting tools
- Measure cutting tools
- Load cutting tools
- Verify program
- Inspect first workpiece
- Run production
- Save corrected program

**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 machining 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.

**Lab exercise (about 5-minutes per student)**

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 (12 minutes)
- read the reading materials for lesson 1.2. (10 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: **37 minutes**

Lesson Plan 1.2 (continued)

**Notes:**

# *Visualizing Program Execution*

**Lesson Plan**

**1.3**

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

Lessons in Key Concept 1 (you are here)

**If supplementing the on-line content with lectures:**

Review content for lessons 1.1 and 1.2. Solicit questions from students.

**Lesson objective:**

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.

- To understand the importance of being able to visualize the execution of a CNC program.
  - 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:**

- The importance of visualizing
- Travel instructions analogy
- Program make-up
- Sequential order of execution
- Machinist vs programmer
  - Machinists advantage
  - Programmer disadvantage
- Job handled by manual machinist
- Job handled by a CNC programmer
- Program structure notes
  - Sequence numbers
  - Word order in a block
  - Decimal point usage
  - Modal words
  - Initialized words
- Common mistakes

**Key points made for each topic:**

**The importance of visualizing – we explain that:**

- Without the ability to visualize a program’s execution, students 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 a student cannot create a set of travel instructions without being able to visualize the path (an analogy is provided in the presentation), neither can a student write a CNC program without being able to visualize how cutting tools will move through their paths.

**Program make-up – we explain that:**

- Programs are made up of blocks. blocks are made up of words. Words consist of a letter address and a numerical value. The letter address specifies the word type.
- Programs are executed in sequential, step-by-step order from beginning to end.

**Machinist versus programmer – we explain that:**

- 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.

**An example job (machinist versus programmer)**

- The presentation describes a simple example job–first done by a machinist, then by a CNC program. It shows the first complete program. While reviewing the program, we explain each block. We point out that students don’t have to memorize the related words and 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 drill machining the hole in their mind, they cannot write the program.

**Program structure notes – We explain:**

- Sequence numbers.
- That the word order within a command is unimportant.
- That certain word types allow a decimal point.
- The meaning of modal.
- The meaning of initialized.
- The most common beginner’s mistakes.

**At the machine (about 20-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.

## Lesson Plan 1.3 (continued)

### Lab exercise (about 5-minutes per student)

Though we have no specific suggestions related to this lesson content, have your students continue practicing with procedures needed to run the machine. As stated, you must be very careful to watch them 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 1.1, students must:**

- view the on-line presentation for lesson 1.3 (14 minutes)
- read the reading materials for lesson 1.3. (10 minutes)
- take the quiz at the end of the reading materials. (5 minutes)

**Exercise**

- Have students take the on-line test for lesson 1.3 (10 minutes)

Approximate total study time for lesson 1.3: **39 minutes**

**Notes:**

## *Understanding the Workpiece Coordinate System*

**Lesson Plan**

# 1.4

*Explains how programmed positions are determined.*

Lessons in Key Concept 1 (you are here)

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine Configurations
- 1.1a: Levil Machining Center Certification Cart
- 1.2: CNC job work flow
- 1.3: Visualizing program execution
- **1.4: Understanding the workpiece coordinate system**
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- 1.7: Introduction to programming words

**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 three dimensional coordinate system from a central origin.

**Main topics for this lesson:**

- Rectangular coordinate system
  - Machine axes
  - Graph analogy
  - 3D coordinate system
- More on polarity
  - XY axes polarity
  - Z-axis polarity
  -
- Where to place program zero
  - in X and Y axes
  - in the Z-axis
- Absolute vs incremental
  - Absolute programming
  - Absolute example
  - Incremental programming
- Inch versus metric programming

**Key points made for each topic:**

**Rectangular coordinate system**
- We begin with an explanation of how an axis drive system works. While students don't need to know the inner workings of a machining 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.
- Next, we show an analogy related to making a graph. The graph is for a company's productivity. We relate each component of a graph to the components of the workpiece coordinate system as it is used for CNC machining 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 (XY). Next, we show the three dimensional coordinate system for a CNC machining center – as well as how points are plotted in X, Y, and Z.

**More on polarity**
- 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 explain that every coordinate used in a CNC program has a polarity (plus or minus).
- We also explain that with coordinates having a positive polarity, the polarity sign (plus) is assumed. Students must only include a polarity sign with negative coordinates (-).

**Where to place the program zero point – we explain 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 placed 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.

**Absolute versus Incremental positioning - we explain that:**
- when coordinates are specified from program zero, it is called the absolute mode of programming.
  G90 specifies absolute positioning mode.
- students should concentrate on absolute positioning.
- another positioning mode is available: the incremental positioning mode.
- with incremental positioning (specified by G91), positions are specified from the cutting tool's last position.
- 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.

**Inch versus metric**
- We point out that with most machines, programs can be developed in either measurement system mode.
- We explain that while most companies in the United States use the inch mode, there is an accuracy advantage to using the metric mode (shown during the slide show).

## Lesson Plan 1.4 (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)

First, demonstrate how the relative position display can be used for taking measurements on the machine. Then have students practice.

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 when measuring program zero assignment values).

**Time to complete lesson 1.4, students must:**

- view the on-line presentation for lesson 1.4 (20 minutes)
- read the reading materials for lesson 1.4. (15 minutes)
- take the quiz at the end of the reading materials. (5 minutes)

**Exercises**

- Have students take the on-line test for lesson 1.4 (10 minutes)
- Have students complete and submit the coordinate sheet exercise for lesson 1.4 (10 minutes)

Approximate total study time for lesson 1.4: **60 minutes**

**Notes:**

## *Determining Workpiece Coordinate System Offsets*

**Lesson Plan**

**1.5**

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

Lessons in Key Concept 1 (you are here)

1: Know Your Machine from a Programmer's Viewpoint

- 1.1: Machine Configurations
- 1.1a: Levil Machining Center Certification Cart
- 1.2: CNC job work flow
- 1.3: Visualizing program execution
- 1.4: Understanding the workpiece coordinate system
- **1.5: Determining workpiece coordinate system offsets**
- 1.6: Setting workpiece coordinate system offsets
- 1.7: Introduction to programming words

**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 should 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:**

- Zero return position
- Program zero assignment values
- Measuring versus calculating
- Retaining for future use

**Key points made for each topic:**

**Program zero must be assigned**

- We explain that when a setup is made, it may be possible for the setup person to make the setup just about anywhere on the machine's table. If program zero is placed at datum surfaces on the workpiece (as it almost always is) the location of program zero within the setup will change based upon the placement of the workholding device.

**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.
- three lights – called *axis origin lights* – will come on when the machine is at its zero return position.

**Program zero assignment values**

- We explain that one way to determine program zero assignment values is to measure them at the machine.
- Students learn that for X and Y, program zero assignment values represent the distances between the spindle center while the machine is at the X and Y zero return position and the X and Y program zero point on the workpiece.
- And for Z, the program zero assignment value is based upon how a feature called tool length compensation is used (discussed in Key Concept number four). If using our recommended methods, the Z axis program zero assignment value is the distance between the spindle nose and the Z axis program zero point on the workpiece.

**Measuring program zero assignment values**

- We explain that to measure program zero assignment values in X and Y, an edge finder or dial indicator can be used.
- We provide to examples of program zero assignment value measurement for X and Y – with an edge finder for a rectangular workpiece and with a dial indicator for a round workpiece.
- We provide one way to measure the program zero assignment value in Z.

**Calculating program zero assignment values**

- We explain that with qualified setups, it may be possible to calculate the location of program zero in each axis (eliminating the need to measure program zero assignment values).

**Retaining program zero assignment values**

- We explain that with qualified setups, and even when you can't calculate program zero assignment values, it is possible to retain the program zero point location for future use. This will also eliminate the need to measure and enter program zero assignment values the next time the job is run.

**Using a spindle probe**

- We point out that a spindle probe facilitates the task of measuring and entering program zero assignment values. The slide show helps you show how a spindle probe is used.

## Lesson Plan 1.5 (continued)

### At the machine (about 20-minutes)

With a workholding setup made on your lab machine, demonstrate the techniques used to measure program zero assignment values. We recommend doing so with a rectangular workpiece using an edge finder.

This demonstration will require procedures to start the spindle (if you're using a "wiggler" style edge finder), jog the axes, use the handwheel, set and reset the relative position displays, and to do a zero return, which have been practiced in previous lessons

### Lab exercise (about 5-minutes per student)

Have students practice measuring program zero assignment values using techniques you have just demonstrated.

Suggested procedures for hands-on practice:

- To measure program zero assignment values in the X and Y axes (included in setup procedures of operation handbook)
- To measure program zero assignment values in the Z axis (included in setup procedures of operation handbook)

**To complete lesson 1.5, students must:**

- view the on-line presentation for lesson 1.5 (12 minutes)
- read the reading materials for lesson 1.5 (25 minutes)
- take the quiz at the end of the reading materials (10 minutes)

**Exercises**

- 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 (15 minutes)

Approximate total study time for lesson 1.5: **72 minutes**

**Notes:**

## *Setting Workpiece Coordinate System Offsets*

**Lesson Plan**

1.6

*Explains how program zero is actually assigned.*

Lessons in Key Concept 1 (you are here)

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine Configurations
- 1.1a: Levil Machining Center Certification Cart
- 1.2: CNC job work flow
- 1.3: Visualizing program execution
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- **1.6: Setting workpiece coordinate system offsets**
- 1.7: Introduction to programming words

**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 workpiece coordinate system offsets (most machines) or with G92 in a program (old machines).

**Main topics for this lesson:**

- Using G92 in the program
- Workpiece coordinate system offsets
- Advantages of workpiece coordinate system offsets

**Key points made for each topic:**

**Assigning program zero in the program (old machines) - we explain that:**

- regardless of how program zero is assigned, the program zero assignment values shown in lesson 1.5 will be used.
- the polarity for program zero assignment values is taken from program zero to the zero return position (almost always positive.
- a G92 command at the beginning of the program includes the program zero assignment values.
- G92 has many limitations and can be difficult – if not dangerous – to use (limitations are shown in the slide show). Workpiece coordinate system offsets eliminate these limitations.
- a zero return command (G28) should be included prior to the G92 command to ensure that the machine is in the proper position prior to executing the G92 command.

**Assigning program zero with workpiece coordinate system offsets (new machines) - we explain that**

- regardless of how program zero is assigned, the program zero assignment values shown in lesson 1.5 will be used.
- With workpiece coordinate system offsets, the polarity for program zero assignment values is taken from the zero return position to program zero (they are almost always negative).
- Fanuc-controlled machines come standard with six fixture offsets – more can be purchased as an option.
- the program zero assignment values are entered into the appropriate fixture offset registers.
- if assigning one program zero point in the program (a common scenario), use fixture offset number one (specified by G54 in the program).
- the advantages of fixture offsets are subtantial (over assigning program zero with G92 in the program).

## Lesson Plan 1.6 (continued)

### At the machine (about 20-minutes)

We’re assuming you are using workpiece coordinate sytem offsets to assign program zero.  Use the program zero assignment values measured in lesson five to show how they are entered into work coordinate system offset registers.  Show the various workpiece coordinate sytem offset pages.

If you have an old machine that doesn’t have workpiece coordinate system offsets, then show how the G92 command in a program must be edited in order to assign program zero.

### Lab exercise (about 5-minutes per student)

Again, we recommend that you provide a step-by-step procedure to enter workpiece coordinate system offsets so students can practice with minimal help from you.  Have them work with an unused workpiece coordinate system offset for practicing (like workpiece coordinate system offset number six so they cannot overwrite needed offset values.

Suggested procedures for hands-on practice:

- To enter workpiece coordinate system offsets (included in manual procedures of operation handbook)

**To complete lesson 1.6, students must:**

- view the on-line presentation for this lesson (4 minutes)
- read the reading materials for this lesson (10 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. (8 minutes)
- Have students complete and submit the coordinate sheet exercise for this lesson (15 minutes)

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

**Notes:**

## *Introduction to Programming Words*

**Lesson Plan**

1.7

*Introduces students to the word types used in CNC programs.*

Lessons in Key Concept 1 (you are here)

1: Know Your Machine from a Programmer's Viewpoint
- 1.1: Machine Configurations
- 1.1a: Levil Machining Center Certification Cart
- 1.2: CNC job work flow
- 1.3: Visualizing program execution
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- **1.7: Introduction to programming words**

**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 machining 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 machining 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.

### 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)

Since students have now been exposed to CNC words, this makes a good time to have student begin working with (editing) programs in the control's memory. Be sure to pick an unneeded program for them to practice on, since it will be modified during this lab exercise.

Suggested procedures for hands-on practice:

- To get ready to edit programs
- To show a directory of programs
- To call up a program from within the CNC memory (make it the active program)

**To complete lesson 1.7, students must:**

- view the on-line presentation for this lesson (18 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 and submit the coordinate sheet exercise for this lesson (15 minutes)

Approximate total study time for this lesson: **58 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
1.1a: Levil Machining Center Certification Cart
1.2: CNC job work flow
1.3: Visualizing program execution
1.4: Understanding the workpiece coordinate system
1.5: Determining workpiece coordinate system offsets
1.6: Setting workpiece coordinate system offsets
1.7: 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: Tool length compensation
4.3: Cutter radius compensation
4.4: Workpiece coordinate system offsets
**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: Hole-machining canned cycles
6.2: Working with subprograms
6.3: Other special programming features
6.4: Programming rotary devices
**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 24 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 machining 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 any clamps or 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 machining 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.
- We also recommend that all Z coordinates be calculated prior to programming – and often several Z coordinates are required for one XY position (consider center-drilling, drilling, and tapping a hole).  We provide a way to easily document these Z coordinates in the slide show.

**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, clamps 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 student manual), 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**

Have students continue practicing editing programs. Again, be sure they are manipulating an unneeded program.

Suggested procedures for hands-on practice:

- To load programs (use your DNC system or memory card from which to load the program)
- To delete programs (again, be sure not to let students delete a needed program)
- To search within a program (the active program)

**To complete lesson 2.1, students must:**

- view the on-line presentation for this lesson (13 minutes)
- read the reading materials for this lesson (25 minutes)
- complete the coordinate sheet 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 and submit the coordinate sheet exercise for this lesson (15 minutes)

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

**Notes:**

Key Concept

3

# Lesson Plan: Understand the Motion Types

*Introduces Key Concept number three.*

**1: Know Your Machine from a Programmer's viewpoint**
- 1.1: Machine Configurations
- 1.1a: Levil Machining Center Certification Cart
- 1.2: CNC job work flow
- 1.3: Visualizing program execution
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- 1.7: 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: Tool length compensation
- 4.3: Cutter radius compensation
- 4.4: Workpiece coordinate system offsets

**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: Hole-machining canned cycles
- 6.2: Working with subprograms
- 6.3: Other special programming features
- 6.4: Programming rotary devices

**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 24 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 1.4). 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 machining centers.

Key Concept 3 (continued)

Notes:

# *Programming the Three Basic Motion Types*

**Lesson Plan**

**3.1**

*Presents the three most common ways a CNC machining 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.
- In X and Y, hole-machining cutting tools (drills, taps, reamers, etc.) are simple. The hole's center line coordinates are programmed (center of the cutting tool).
- In Z, hole-machining cutting tools usually require the programmer to compensate for the cutting tool's lead. That is, it is the very tip of the cutting tool in Z that is being programmed.
- In X and Y for milling cutters, we point out that sometimes it is the center of the cutting tool that is programmed – especially when face milling. But when side-milling, and especially when milling a contour, there is a feature called cutter radius compensation (presented in Key Concept number four), that allows the programmer to specify work-surface coordinates in the program – which makes programming much simpler. But we point out that for now, the centerline positions for milling cutters must be used for all examples (until Key Concept number four is shown).

**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)

**Full circle in one command**

- We point out that one exception to the "crossing one quadrant line" limitation is commanding a full circle in one command. We show the technique for doing so.

**Helical interpolation**

- We introduce a fourth motion type called *helical interpolation*. This motion type is used for thread milling.
- We point out that if a company does not perform thread milling operations, there will be no need for helical motion.
- We explain that thread milling (including helical motion) is presented in Key Concept number six.

### At the machine (20-30 minutes)

Students actually 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 a fixture offset.

While you haven't presented this yet, point out that cutting tools must be assembled and measured – and that tool lengths must be entered into tool length compensation offsets (these techniques are shown in Key Concept number four).

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

Have students continue practicing editing programs. Again, be sure they are manipulating an unneeded program.

Suggested procedures for hands-on practice:

- To alter, insert, and delete (have them practice altering words, inserting words and commands, and deleting words and commands)

**To complete lesson 3.1, students must:**

- view the on-line presentation for this lesson (21 minutes)
- read the reading materials for this lesson (35 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: **136 minutes**

**Notes:**

Key Concept

**4**

## Lesson Plan: *Understand the Motion Types*

*Introduces Key Concept number four.*

**1: Know Your Machine from a Programmer's viewpoint**

- 1.1: Machine Configurations
- 1.1a: Levil Machining Center Certification Cart
- 1.2: CNC job work flow
- 1.3: Visualizing program execution
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- 1.7: 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: Tool length compensation
- 4.3: Cutter radius compensation
- 4.4: Workpiece coordinate system offsets

**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: Hole-machining canned cycles
- 6.2: Working with subprograms
- 6.3: Other special programming features
- 6.4: Programming rotary devices

**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 24 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: Tool length compensation
- 4.3: Cutter radius compensation
- 4.4: Workpiece coordinate system offsets

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

Key Concept 4 (continued)

**Notes:**

# *Introduction to Compensation*

**Lesson Plan**

4.1

*Explains why compensation is required with CNC machining centers.*

Lessons in Key Concept 4 (you are here)

4: Know the compensation types

- **4.1: Introduction to compensation**
- 4.2: Tool length compensation
- 4.3: Cutter radius compensation
- 4.4: Workpiece coordinate system offsets

**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. Neither will they know the precise length of each cutting tool nor possibly the precise diameters of milling cutters.

**Lesson objective:**

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

**Main topics for this lesson:**

- Marksman analogy
- Understanding compensation
  - Tool length compensation
  - Cutter radius compensation
  - Fixture offsets
- 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 machining 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.
  - 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 tool lengths, milling cutter radii, and fixture placement).
- We describe the various offset display screen pages on a typical CNC machining 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

Have students continue practicing editing programs. Again, be sure they are manipulating an unneeded program.

Suggested procedures for hands-on practice:

- To save programs (have them save programs to a memory card or you DNC system)
- To use background edit

**To complete lesson 4.1, students must:**

- view the on-line presentation for this lesson (11 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: **81 minutes**

**Notes:**

# *Tool Length Compensation*

**Lesson Plan**

4.2

*Presents tool length compensation, which is used in every tool of every program.*

Lessons in Key Concept 4 (you are here)

4: Know the compensation types

- 4.1: Introduction to compensation
- **4.2: Tool length compensation**
- 4.3: Cutter radius compensation
- 4.4: Workpiece coordinate system offsets

**If supplementing the on-line content with lectures:**

Point out that cutting tools vary in length. And it is difficult (if not impossible) to know the precise length of each cutting tool as a program is written. Indeed, programs are often prepared long before they are needed at the CNC machine. The feature tool length compensation will allow the programmer to write programs without knowing the lengths of cutting tools.

When working on practice programs, students have seen the command to instate tool length compensation (the G43 command). They may have questioned you about this command already. Explain that in this lesson, we'll be presenting the details of how tool length compensation is used.

**Lesson objective:**

*To help students understand tool length compensation, including how it is programmed as well as how it is used at the machining center.*

**Main topics for this lesson:**

- Reasons for tool length compensation
- Two ways to use
  - Offset is tool length
  - Offset is distance from tool tip to Z zero
- Typical mistakes
- Sizing and trial machining

**Key points made for each topic:**

**Reasons for tool length compensation**

- We present the reasons why tool length compensation is required: cutting tools can be of any length, sizing and tool wear adjustments for depth attributes (like pocket depths), and trial machining for depth attributes.
- We point out that tool length compensation is required for every tool in every program, meaning that students must thoroughly understand and master this feature.

**Two ways to use**

- We include the two most popular methods to use tool length compensation.
- To avoid confusion, we recommend that students only use one of these methods.
- We explain that, generally speaking, companies that have a lot of repeated jobs (like product-producing companies) often have several people involved with the tasks related to making setups. These companies tend to prefer the first method (tool length is the offset value).
- We also explain that companies having but one person performing all of the tasks related to making setups (like many contract shops) tend to use the second method (distance from tool tip to program zero is the offset value).
- We point out that the our recommended shown has several advantages over the other: tool length compensation values can be determined off line, tool length compensation values for tools used in consecutive jobs need not be re-measured or re-entered, cutting tools can be used on different machines with the same tool length compensation values, and multiple identical cutting tools can be kept ready for action. We strongly recommend that students use the first method – where the tool length compensation value is the tool's length.

**Offset is the tool's length**

- We illustrate that with this method, the program zero assignment value in Z (fixture offset Z value) is the distance from the spindle nose to the Z axis program zero point (a negative valueWe also also show how to measure this value right on the.
- We also explain that the tool length compensation value (the tool length compensation offset value for the tool) is the distance from the tool tip to the spindle nose (a positive value). We show how this value can be measured right on the machine and then entered into the tool length compensation offset.
- Next, we show how tool length compensation values can be measured off line, in preparation for an up-coming job. We even show how a special program can be created that enters tool length compensation values into offset registers.

**Offset is distance from tool tip to program zero in Z**

- Again, we recommend that students use the other method. This is not our recommended method.
- We illustrate that the program zero assignment value in Z (the fixture offset Z) will be zero when using this method.
- We also illustrate what the offset value represents if this method is used – as well as how to measure and enter it.

**Programming** tool length compensation

- We point out that programming remains exactly the same regardless of which method (shown above) is used. The only differences are related to the fixture offset Z value and the value stored in the offset – both of which are unrelated to programming.

## Lesson Plan 4.2 (continued)

- We explain the words involved with tool length compensation, including G43 and the H word. Also, explain that tool length compensation is instated during each tool's first Z axis movement (its approach movement).
- We explain that the offset number used with each tool should in some way be made to correspond to the tool station number. We recommend using the same offset number as tool station number (tool one – offset one, tool two – offset two, and so on).
- We show an example program to illustrate the programming of tool length compensation.
- We explain that since tool length compensation remains in effect until the next tool's instating command, there is no need to cancel it (though there is a G code to do so – G49).

**Typical mistakes**

- We warn students about mistakes they'll be prone to making – forgetting to instate it and forgetting to enter offset values. We also show the consequences of these mistakes.

**Sizing and trial machining**

- This is of special importance to setup people and operators. We point out that some depth dimensions are critical (having a very small tolerance).
- We explain that when a setup person is worried that an initial tool length compensation setting isn't accurate enough (this is commonly the case when a depth tolerance is less than about 0.001 inch or so), they can use trial machining to ensure that the cutting tool will not remove too much material (scrapping the workpiece) the first time it machines.
- We present the five steps to trial machining.
- We point out that even with more open tolerances, it is commonly necessary to make a tool length compensation offset adjustment after machining the first workpiece. While the machined attribute will be within the tolerance band on the first workpiece, it may not be right at the target value. An adjustment can be made to ensure that the attribute will be right at the target value for the *next* workpiece to be machined.
- We explain that students must understand that as cutting tool's wear, they tend to leave more material on the surfaces they machine. If tolerances are tight, a machined surface may shrink or grow out of its tolerance band before the cutting tool gets dull. If it does, a sizing adjustment (possibly several of them) will be necessary during the cutting tool's life.
- We explain that eventually cutting tools will wear out and must be replaced. If trial machining was needed for a cutting tool during setup, it will be necessary every time the cutting tool is replaced.

### At the machine (about 45 minutes)

There are several things you can do at the machine to demonstrate the principles presented in this lesson.

- Using a step-by-step procedure made for your machine, demonstrate tool length compensation offset measurement right on the machining center/s in your lab. Be sure to include how tool length compensation values can be transferred directly into offsets right from the relative display screen page.
- If you have an off-line tool length measuring device, demonstrate its use. Be sure to show how tool length compensation offsets are entered into the machine.
- Students will be working on a program in the programming activity for this lesson. Use it – or one of your own – to have them assemble the related cutting tools and take turns measuring them on the machine (using your step-by-step procedure for doing so). As always, watch them carefully to avoid mistakes. When they're finished, run the program for them. Point out variations in the Z axis positioning based upon the length of each tool being used (shorter tools will cause the Z axis to move further down on a vertical machining center).

### Lab exercise

While you must continue to watch students carefully, you should have them practice assembling and measuring the lengths of a few cutting tools. Have them write down each tool length compensation value and submit them for grading.

Suggested procedures for hands-on practice:

- To measure and enter tool length compensation values (shown in the setup procedures)
- To enter offset values GEOM(H), WEAR(H) (shown in manual procedures)

**To complete lesson 4.2, students must:**

- view the on-line presentation for this lesson (20 minutes)
- read the reading materials for this lesson (25 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: **100 minutes**

**Notes:**

## *Cutter Radius Compensation*

**Lesson Plan**

4.3

*Presents cutter radius compensation, which is only used with side-cutting milling cutters.*

Lessons in Key Concept 4 (you are here)

4: Know the compensation types
- 4.1: Introduction to compensation
- 4.2: Tool length compensation
- **4.3: Cutter radius compensation**
- 4.4: Workpiece coordinate system offsets

**If supplementing the on-line content with lectures:**

Point out that just as tool length compensation allows the programmer to write programs without knowing each cutting tool's length, so does cutter radius compensation let them do so without knowing each milling cutter's precise radius.

Back in lesson 3.1 (motion commands), students worked on two programs that involved milling cutters (one milling cutter in each program). Remind them that at that time they had to calculate the cutter's centerline tool path – and this complicated the task of calculating coordinates. Every coordinate required them to consider the precise size (radius) of the milling cutter being used.

With cutter radius compensation, the work surface path can be used to determine coordinates, making calculating coordinates much simpler.

**Lesson objective:**

*To help students understand cutter radius compensation, including how it is programmed as well as how it is used at the machining center.*

**Main topics for this lesson:**

- Reasons for cutter radius compensation
- Two ways to use
  - Offset is cutter radius
  - Offset is deviation from planned cutter size
- Steps to programming
  - Instate
  - Machine surfaces
  - Cancel
- Examples
- Sizing and trial machining

**Key points made for each topic:**

**Reasons for cutter radius compensation**
- We point out that cutter radius compensation is only used for milling cutters – and only when the cutter is machining on its periphery (side milling). This feature is never used for drills, taps, or reamers. We mention that some companies don't do much side milling, meaning they don't have much of a need for cutter radius compensation.
- We present the reasons why cutter radius compensation is required: it simplifies the calculations needed for the program, it allows a range of cutter sizes, it allows sizing and trial machining for milled surfaces, and the same set of coordinates used to finish the surfaces being machine can be used for roughing.
- We point out that two of these reasons (simplified calculations and rough and finish milling with the same coordinates) apply only to manual programmers.

**Two ways to use**
- We show the two most popular methods to use cutter radius compensation. To avoid confusion, we recommend that you show only one of these methods, the one preferred by manual programmers.
- We explain that, generally speaking, manual programmers (those not using a CAM system) prefer to program the work surface path and the offset value used with cutter radius compensation will be the radius of the milling cutter being used. We recommend that students use this method.
- We explain that computer aided manufacturing (CAM) system programmers often prefer to let the CAM system generate a tool path based upon the milling cutter's centerline path. The offset value is the radial difference between the planned cutter size (the size the programmer expects the setup person to use) and the milling cutter actually being used.

**Three steps to using**
- We illustrate the three steps to programming cutter radius compensation – instate it, machine the required surfaces, and cancel it. The most difficult step is instating. We provide several examples to help students understand.
- For instating, we explain the two related G codes – G41 and G42. One easy way to explain (if students have basic machining practice experience) is that climb milling requires G41 to instate and conventional milling requires G42. We also explain G41 and G42 even if students don't know the difference between climb and conventional milling.
- Still part of instating, we explain that an offset must be invoked (with a D word) by the instating command. We explain the two most common offset pages of Fanuc controls. With one of them, there is only one offset register per offset. With this style of offset table, the offset corresponding to the tool station number is already being used with tool length compensation. We explain that another offset must be chosen – we recommend adding a constant number (like 30) to the tool station number to come up with the offset number used with cutter radius compensation. With the other offset table, each offset has two registers – one for the tool's length and the other for its radius. With this offset table, have students simply use the offset number corresponding to the tool station number.
- Still for instating, we explain the "prior position" – the XY location just before cutter radius compensation is instated.
- For step two, machining surfaces, we show several examples – and provide some limitations and warnings.

## Lesson Plan 4.3 (continued)

- For canceling (step three), we provide our recommended method to cancel cutter radius compensation and show some examples.

**Examples**

- We show three complete examples of cutter radius compensation.

**Sizing and trial machining**

- This is of special importance to setup people and operators. We point out that just like tool length compensation allows the setup person or operator adjust depth attributes, so does cutter radius compensation allow them to adjust XY milled surfaces. Many of the same principles presented in lesson 4.2 apply to cutter radius compensation.
- We let students know that making offset adjustments with cutter radius compensation offsets can be confusing. The offset contains the milling cutter's *radius*. Often a milling cutter will mill completely around a workpiece, meaning the measurement being taken is over two milled surfaces. This requires the offset to be modified by *half* the deviation.
- We present trial machining and sizing for cutter radius compensation.
- Also as with tool length compensation, we explain that the size of the tolerance determines whether trial machining is necessary whenever a new milling cutter is used – and whether any sizing adjustments will be necessary during the tool's life.

**Roughing with finishing coordinates**

- We illustrate how one set of coordinates (the finishing coordinates) can be used for both roughing and finishing. We point out that the setup person will "lie" to the control about the size of the rough milling cutter's radius, making it larger than the cutter's actual radius. The amount larger will be how much finishing stock the roughing cutter leaves. For example, if using a 1.0 inch diameter cutter (0.5 radius) and if you want to leave 0.03 for finishing, the rough milling cutter's offset must be set to 0.53.

### At the machine (about 15 minutes)

There are several things you can do at the machine to demonstrate the principles presented in this lesson.

- Using a step-by-step procedure made for your machine, demonstrate the entry of cutter radius compensation offsets.
- Students will be working on a programming activity in this lesson. Use it – or one of your own – have them assemble the related cutting tools and take turns measuring their lengths on the machine (using your step-by-step procedure for doing so). Have them also measure the radius of each milling cutter and enter it into the appropriate offset register. As always, watch them carefully to avoid mistakes. When they're finished, run the program for them. Point out the variations in the XY axis motions based upon the size of milling cuter being used (the value in the offset).

### Lab exercise

You can students practice measuring the radius of a few milling cutters. Have them write down each cutter radius and submit them for grading.

Suggested procedures for hands-on practice:

- To enter offset values GEOM(D), WEAR(D) (shown in manual procedures)

**To complete lesson 4.3, students must:**

- view the on-line presentation for this lesson (26 minutes)
- read the reading materials for this lesson (30 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: **121 minutes**

**Notes:**

## *Workpiece Coordinate System Offsets*

**Lesson Plan**

4.4

*Extends what students know about fixture offsets from 1.6*

Lessons in Key Concept 4 (you are here)

4: Know the compensation types

- 4.1: Introduction to compensation
- 4.2: Tool length compensation
- 4.3: Cutter radius compensation
- **4.4: Workpiece coordinate system offsets**

**If supplementing the on-line content with lectures:**

Most of what students need to know about workpiece coordinate system offsets is presented in lesson 1.6. Indeed, if only one coordinate system is required per program (which is commonly the case), it is possible to skip this lesson if you're running short on time.

Students know that program zero must be assigned before a program can be run – and workpiece coordinate system offsets are used to assign program zero. To this point, only one program zero point (coordinate system) has been required per program, and students know how to assign it. In this lesson, we'll be showing how to assign multiple program zero points – and we'll show some time-saving techniques that will apply if setups are qualified and/or predictable.

**Lesson objective:**

*To complete the students' understanding of* workpiece coordinate system *offsets.*

**Main topics for this lesson:**

- Review of primary use
- Assigning multiple program zero points
  - The two ways of entering values
- Working with sub-plates
- Repeated setups
- Programming workpiece coordinate system offset entries

**Key points made for each topic:**

**Review of workpiece coordinate system offset use**

- We review the main points from lesson six – showing how program zero is assigned when only one coordinate system is required in the program.

**Multiple program zero points**

- We show two ways of handling multiple program zero points. The first is used when there is no relationship from one program zero point to another, as would be the case when two or more vises are used to hold workpieces machined by one program. With this method, assigning multiple program zero points is simply an extension of assigning one program zero point. Each set of program zero assignment values is measured and entered just as for one program zero point – there's simply more work for the setup person to do.
- The second method – which is also nicely illustrated – is used when the setup person knows the distances from one program zero point to the next. This method involves using the common workpiece coordinate system offset, which shifts the point of reference from the zero return position to a more logical location within the setup (commonly one of the program zero points used for one of the programs.

**Shifting the point of reference**

- This discussion is actually part of the second method for using workpiece coordinate system offsets just mentioned. We point out that the zero return position doesn't always make a very logical point of reference for fixture offset entries. The method described above is one such time – is the next topic – working with sub-plates.

**Working with sub-plates**

- We present another excellent application for shifting the point of reference for fixture offset entries. Sub-plates are commonly used on vertical machining centers to minimize workholding setup time. But with a little ingenuity, they can also eliminate all of the tasks related to assigning program zero

**Repeated setups**

- We point out that companies that have a lot of repeated jobs strive to qualify their setups. This ensures that the workholding device will be placed on the machine table in exactly the same location every time the setup is made – which also means the program zero location is in the same location every time the setup is made. If this is done, the program zero assignment values used the last time the setup was made will work the next time the setup is made.
- We explain that one way to retain program zero assignment values from one time a job is run to the next is to write them down.
- We also explain that a more efficient way is to include program zero assignment values in the program. That is, commands to assign program zero (enter fixture offsets) can be included in the program with G10 commands. We illustrate how G10 is used.

**Common scenarios**

- Admittedly, these discussions may be confusing to students. Which method should they use? We provide some scenarios to help illustrate which method of program zero assignment is best for a given application.

**How many workpiece coordinate system offsets do you have?**

- Point out that Fanuc controls come standard with six workpiece coordinate system offsets (which is usually enough).

**Lesson Plan 4.4 (continued)**

**Running out?**

- Again, we point out that Fanuc controls come standard with six fixture offsets (which is usually enough). But more can be purchased as an option.

**At the machine (about 10 minutes)**

Call up the workpiece coordinate system offset page and point out the common offsets, which normally has all registers set to zero. Again, point out that the values in this workpiece coordinate system offset specify how far from the machine's zero return position is the point of reference for workpiece coordinate system offset entries – when set to zero, the zero return position *is* the point of reference for workpiece coordinate system offset entries.

- You might want to plug in some values and run an example program (or make some manual data input commands) to demonstrate what will happen when the point of reference for fixture offset entries is shifted.

**Lab exercise**

Review the procedures required for measuring program zero assignment values.

Suggested procedures for hands-on practice:

- To mount the workholding device on the table (shown in setup procedures)
- To measure program zero assignment values in the X and Y axes (shown in setup procedures)
- To measure program zero assignment values in the Z axis (shown in setup procedures)
- To enter workpiece coordinate system offset values (shown in manual procedures)

**To complete lesson 4.4, 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 (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: **85 minutes**

**Notes:**

**Key Concept**

**5**

## *Lesson Plan: You Must Provide Structure to Your CNC Programs*

*Introduces Key Concept number five.*

- **1: Know Your Machine from a Programmer's viewpoint**
  - 1.1: Machine Configurations
  - 1.1a: Levil Machining Center Certification Cart
  - 1.2: CNC job work flow
  - 1.3: Visualizing program execution
  - 1.4: Understanding the workpiece coordinate system
  - 1.5: Determining workpiece coordinate system offsets
  - 1.6: Setting workpiece coordinate system offsets
  - 1.7: 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: Tool length compensation
  - 4.3: Cutter radius compensation
  - 4.4: Workpiece coordinate system offsets
- **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: Hole-machining canned cycles
  - 6.2: Working with subprograms
  - 6.3: Other special programming features
  - 6.4: Programming rotary devices
- **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 24 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 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.
- We name a few programming words: G00, G01, G02, G03, M03, the O word, F word, S word – and so on - to see if students can describe them. It’s likely that they can. We point out that they already know quite a bit about the structure related to writing CNC programs.
- We remind students that learning to write CNC programs is like learning a foreign language that has only about 50-60 words. They’ve already been exposed to 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 many things are dictated by the way programs are formatted. And it's best if *we* control the objectives that programs achieve (safety, ease of use, efficiency, etc.).

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.
- We explain that all programs shown in this class stress the most popular kind of automatic tool changer – the double arm style.
- We also present the single arm automatic tool.

**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 block 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. If you have a double-arm automatic tool changer, be sure to point out the difference between rerunning a tool that is already in the spindle (restart block is the command after the M06) and rerunning a tool that is not currently in the spindle (restart block is the M06 command.

**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 5.1, students must:**

- view the on-line presentation for this lesson (7 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: **77 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
- Format for machining centers using fixture 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.
- We provide presentation about G28 (the zero return command) – including several examples and warnings. This command tends to be the most confusing and misunderstood Fanuc G code.

**Program formats**

- We provide four sets of format: for vertical and horizontal machining centers with and without workpiece coordinate system offsets. To avoid confusion, have students view only the one that applies best to your lab machine/s (probably for vertical machining centers using workpiece coordinate system offsets).
- 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, M30, and M19). 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

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 5.2, students must:**

- view the on-line presentation for this lesson (12 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)
  - 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: **87 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
  - 1.1a: Levil Machining Center Certification Cart
  - 1.2: CNC job work flow
  - 1.3: Visualizing program execution
  - 1.4: Understanding the workpiece coordinate system
  - 1.5: Determining workpiece coordinate system offsets
  - 1.6: Setting workpiece coordinate system offsets
  - 1.7: 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: Tool length compensation
  - 4.3: Cutter radius compensation
  - 4.4: Workpiece coordinate system offsets
- **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: Hole-machining canned cycles
  - 6.2: Working with subprograms
  - 6.3: Other special programming features
  - 6.4: Programming rotary devices
- **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 24 lessons.

Key Concept number six is four-lesson key concept.

**Points made during the introduction to Key Concept number six**

- At this point, students are 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 drilling 50 holes using only G00 and G01 will take at least 150 commands).
- In Key Concept number six, we'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
- Point out that while it is important for students to understand the various special programming features that are available, not all will be of immediate importance. What may be quite helpful and often-used for one programmer may never be needed by another. We'll be placing an emphasis on the most popular features

**Lessons for this Key Concept**

- **6: Special features that help with programming**
  - 6.1: Hole-machining canned cycles
  - 6.2: Working with subprograms
  - 6.3: Other special programming features
  - 6.4: Programming rotary devices

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

**Notes:**

# Hole-Machining Canned Cycles

**Lesson Plan**

**6.1**

*Helps students master the use of hole-machining canned cycles.*

Lessons in Key Concept 6 (you are here)

6: Special features that help with programming

- **6.1: Hole-machining canned cycles**
- 6.2: Working with subprograms
- 6.3: Other special programming features
- 6.4: Programming rotary devices

**If supplementing the on-line content with lectures:**

Solicit questions about previous topics.

Student know how to program the drilling of holes using G00 and G01. Now you'll show them how to easily program any hole machining operation – with but one command required per hole.

- Once the first hole is programmed, programming other holes of the same type is as simple as listing hole coordinates.
- When finished programming a series of holes, the cycle is cancelled (with G80).
- 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:**

*To help students master the programming of any hole machining operation.*

**Main topics for this lesson:**

- Introduction to canned cycles
  - Two things they all have in common
- Description of each canned cycle
- Understanding G98 and G99
- Example
- Canned cycles in the incremental mode

**Key points made for each topic:**

**Introduction to canned cycle commonalities**

- We explain what a "canned cycle" is (a series of pre-determined motions specified by one program command). We review G28 command, pointing out that it is a kind of canned cycle. One command actually does two things (the motion to the intermediate position and then the motion to the zero return position).
- We point out that canned cycle share two things in common. First, they are all modal. Once a canned cycle is instated, it will remain in effect until it is cancelled (with G80).
- Second, all hole-machining canned cycles will cause four basic motions: 1) rapid to XY position. 2) rapid to R plane. 3) machine the hole. 4) retract from the hole. The slide show helps you illustrate these four motions.

**Description of each canned cycle**

- We show the exact motions for each of the canned cycle in order of popularity. That is, we show the most popular canned cycles first.
- We introduce all of the words related to each canned cycle. We point out the consistency for word use from one canned cycle to another (especially X, Y, R, Z, and F).

**Simple example**

- We present a simple example to help students understand the points you've made so far.

**Understanding G98 and G99**

- We explain the initial and R planes. This can be a little difficult for beginners to master. We point out that G98 and G99 (initial plane retract and R plane retract) allow the programmer to easily clear clamps and other obstructions (in Z) between holes.
- We explain that G98 specifies that, after the hole is machined, the tool will retract to the *initial plane*. The initial plane is defined as the tool's last Z position prior to the canned cycle command. G99 specifies that the tool will retract to the rapid plane (specified in the canned cycle command itself.
- We explain that the initial plane is usually specified right in the tool's first Z axis approach motion (the G43 command). If clamps are to be avoided, this Z position must be above the clamps (usually 2.0 inches above the work surface is sufficient.
- We point out that if no obstructions exist between holes, the programmer can rapid the tool right to the R plane. In this case the rapid plane and the initial plane will be the same.
- We like to see students placing the G98 or G99 word at the end of the command to help remind them that these words control what happens *after* the hole is machined.
- We explain that G98 and G99 are modal. They are canceled when the G80 word (canned cycle cancellation) is executed. G98 is the initialized state (and reinstated at G80), meaning obstructions will be avoided if no G99 or G98 is specified.
- We provide several examples that illustrate the use of G98 and G99.

**Canned cycles and the Z axis**

- We point out that the R and Z word in the canned cycle command are absolute values (if working in the absolute mode – G90). We illustrate how to program holes into different Z surfaces.

**Using the incremental mode**

- We let students know that if they have a series of equally spaced holes to machine (like a grid pattern of holes), it may be easier to program them using the incremental mode. We explain that the meanings of X, Y, R, and Z change in the incremental mode.
- We explain that an L word in the canned cycle command specifies the number of holes to be machined by the command.

## Lesson Plan 6.1 (continued)

### At the machine (about 20 minutes)

In the programming activity for this lesson, students will be writing a program that use canned cycles. You can easily use it (or exercises of your own design) for practice at the machine. Be sure to illustrate how easy it is to modify canned cycle commands as the need arises (like changing G81 to G73 and adding a Q word to change from standard drilling to chip-break peck drilling.

Be sure to illustrate all of the most popular cycles (G73, G81, G82, G84, and G86).

### Lab exercise

Since the program written during this lesson contain several tools – and since canned cycles make the program so much shorter – you should be able to continue stressing 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 counter-bored hole has been machined by a tool in station number five and it is too shallow by 0.005 inch? Ask which offset is involved and how it must be modified to make the tool go 0.005 deeper.

Suggested procedures for hands-on practice:

- To rerun tools

**To complete lesson 6.1, students must:**

- view the on-line presentation for this lesson (13 minutes)
- read the reading materials for this lesson (30 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 second 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: **98 minutes**

**Notes:**

## *Working with Subprograms*

**Lesson Plan**

**6.2**

*Presents the applications and use of subprograms*

Lessons in Key Concept 6 (you are here)

6: Special features that help with programming

- 6.1: Hole-machining canned cycles
- **6.2: Working with subprograms**
- 6.3: Other special programming features
- 6.4: Programming rotary devices

**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 presented two times when commands in a program must be repeated: multiple operations on holes (center drill, drill, and tap, for example) and when rough and finish contour milling. 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 hole machining operations
  - Rough and finish side milling
  - Multiple identical pockets
- 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.
- We show the application for multiple operations on holes – nicely illustrating how helpful a subprogram can be.
- We shoe the application categories, and list example applications in each. Later in the presentation we show complete examples.

**Related words**

- We then introduce the four words used with subprograms (M98, M99, P, and L).
- Next we present a simple example to help students understand the points 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 the G52 command, which allows temporary shifting of the program zero point. This is required to eliminate the need to program the operation in the incremental positioning mode. We introduce and explain G52.
- The *control programs* application example is related to machining centers that have pallet changers. But of course, not all machining centers have this accessory. However, this example will help students understand control programs.

**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.

### At the machine (about 10 minutes)

In the programming activity for this lesson, students will be writing program that use subprograms. You can easily use it (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 seems 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 6.2, 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 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: **73 minutes**

**Notes:**

## *Other Special Programming Features*

**Lesson Plan**

# 6.3

*Shows 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: Hole-machining canned cycles
- 6.2: Working with subprograms
- **6.3: Other special programming features**
- 6.4: Programming rotary devices

**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 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 but never needed by another. But before students can begin to apply a feature, of course, they must know it exists.

- Optional block skip is a feature needed and used by most companies.
- Statement labels are rarely needed.
- Helical motion is only needed if the company performs thread milling operations.

You can, of course, skip or minimize presentations for topics you know students will not need.

**Lesson objective:**

*To help students recognize and master those special programming features that are important to them.*

**Main topics for this lesson:**

- Optional block skip techniques
- N words as statement labels
- 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 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.
- G02-G03: We provide a good explanation of helical motion as it is used when thread milling.
- 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).
- 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. Here we show some techniques for the *data setting command*.
- G15, G16: We then show how polar coordinates are programmed. We introduce polar coordinates and show limitations for this feature.
- G17, G18, G19: We present the implications of plane selection. Examples include milling with a ball end mill and using right-angle heads.
- 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.
- G30: We show how some, but very few, machining centers use the secondary reference point as the tool change position in the Z axis (instead of the zero return – G28 – position).
- G50, G51: We introduce the feature scaling. Since it is so rarely used, we simply introduce it and show the related words.
- G60, G64: We explain the application and use for single-direction positioning (used when finish boring holes).
- G68, G69: We explain coordinate rotation. A good example is provided.
- G94, G95: We explain that some, but not many, CNC machining centers allow feedrate specification in feed per revolution as well as in feed per minute. For those that do, these two G codes are involved (G94 for per-minute and G95 for per-revolution).
- G50.1, G51.1: We explain the applications and limits of *mirror image* and show examples.

**Other M codes of interest**

- We explain that 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: We explain that some, but not many, machining centers allow the spindle and coolant to be activated at the same time.
- If you know that the machine/s your students will be using have M codes not addressed by this class, you should introduce and explain them here.

## Lesson Plan 6.3 (continued)

**Understanding parameters**

- We explain that 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.

### 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 may be options, meaning you’ll have to limit your program to showing only those features that are equipped on your machine.

With the machine well off its over-travel limits in each axis yet clear of obstructions, for example, you can easily demonstrate helical motion with the MDI command (it makes a 2.0 inch diameter motion):

```
G91 G02 I-1.0 Z-1.0 F20.0
```

### Lab exercise

The programming activity of this lesson involves writing a program using helical motion to machine a thread. If you have students do this exercise, be sure to 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 (17 minutes)
- read the reading materials for this lesson (25 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 it (20 minutes)
  - Have students submit the completed (typed) program for grading

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

**Notes:**

## *Programming Rotary Devices*

**Lesson Plan**

**6.4**

*Explains how the fourth and fifth axes are programmed.*

Lessons in Key Concept 6 (you are here)

6: Special features that help with programming
- 6.1: Hole-machining canned cycles
- 6.2: Working with subprograms
- 6.3: Other special programming features
- **6.4: Programming rotary devices**

**If supplementing the on-line content with lectures:**

Solicit questions about previous topics.

Students now know how to handle the three linear axes: X, Y, and Z. Explain that when a fourth (and possibly fifth) axis is added to a machining center, it is usually a *rotary* axis. And there are substantial differences related to the way these axis types are handled.

Point out that with the programming methods you've shown so far, only one side of the workpiece has been exposed to the spindle for machining – and probably only one program zero point has been required (per workpiece). When equipped with a rotary device, the machine can rotate the workpiece to expose several sides to the spindle for machining during the CNC cycle.

- Programs for machines with rotary devices tend to be longer – but no more difficult to write.
- While there are similarities to linear axes, there are also substantial differences.

You can, of course, skip – or minimize presentations for– topics in this lesson if your machining centers do not have rotary devices.

**Lesson objective:**

*To help students master the programming of a rotary axis.*

**Main topics for this lesson:**

- Indexers versus rotary axes
- Programming indexers
- Programming rotary axes
  - Comparison to linear axes
- Example program

**Key points made for each topic:**

**Introduction**
- We begin by introducing the two types of rotary devices – indexers and rotary axes.
- We explain that an indexer can only rotate to expose a surface for machining. No machining can occur during the rotation.
- We explain that a rotary axis can be used as an indexer, but it also allows machining to occur during rotation. That is, the feedrate for rotation can be precisely specified.

**Programming indexers**
- We explain the programming of three indexer types (90 degree indexers, 5 degree indexers and 1 degree indexers).

**Programming a rotary axis**
- We explain the programming of a full rotary axis (limit your presentation, of course, to the kind/s of rotary devices equipped on your machine/s.
- We explain how the letter address (A, B, or C) is chosen based upon rotary axis orientation with the spindle.
- We explain rotary axis polarity.
- We explain that a rotary axis, just like a linear axis, has a zero return position. We illustrate this position and how it is programmed with G28.
- We explain that as opposed to a linear axis, a rotary axis has no over-travel limits. It can continue rotating an unlimited number of times.
- If a rotary axis is being used as an indexer (rotation then machining), we recommend programming it in the incremental mode. We do explain the assignment of program zero for a rotary axis.

**Programming a rotary axis**
- We explain both absolute and incremental programming. But again, we recommend programming most rotations in the incremental mode.
- We explain the use of G00 for indexing (rotation at rapid) as well as G01 for cutting motions.
- We explain the use of feedrate (F word) when G01 is used for rotation. Feedrate must be specified in *degrees* per minute (not inches per minute). We show how to calculate degrees per minute feedrate.
- Though it is not described in detail, we do mention the feature *cylindrical interpolation* as well as its application.

**Approaching applications**
- We present the approach for handling rotary device programming. Included are processing considerations, helping students decide where to place a central program zero point, assigning multiple program zero points, and when to change tools.

**Example program**
- We provide a sample program in the presentation, but the reading materials show a more realistic example. These programs should help students understand the programming of a rotary device.

## Lesson Plan 6.4 (continued)

### At the machine (about 20 minutes)

Be sure to demonstrate the “unusual” things that can happen in the absolute mode. I like to give the command G90 G00 B0 first (in MDI). (Of course, use the letter address that applies to *your* rotary axis.) The rotary axis will rotate to the zero side. Then give the command G00 B90.0. The axis will rotate ninety degrees. Then G00 B180.0 – ninety more degrees. Then G00 B275.0 – ninety more degrees. Next type G00 B0, but before executing, ask students what will happen.

To their surprise, the axis will rotate 270.0 degrees in the opposite direction. This demonstration nicely illustrates that when going from a small B value to a larger one, the rotary axis will rotate in the positive direction (normally clockwise when viewed from above the axis). But when going from a large B to a smaller one, the axis will rotate in the negative (counter-clockwise direction).

You might also demonstrate the G01 mode – as well as how feedrate is specified in degrees per minute. In MDI, execute the command G91 G01 B90.0 F90.0. Have students note the large F word (F90.0). This motion will take a full minute to execute (ninety degrees of rotation at ninety degrees per minute.

### Lab exercise

In the exercise for this lesson, students write a program that can be used to demonstrate the rotary axis. If you have a machining center with a rotary axis, have students load and run it. Note that multiple program zero points must be assigned prior to running the program.

**To complete lesson 6.4, students must:**

- view the on-line presentation for this lesson (7 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 (20 minutes)
    - If using NC Guide, have students type the program with NC Guide and verify the it (20 minutes)
    - Have students submit the completed (typed) program for grading

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

**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
- 1.1a: Levil Machining Center Certification Cart
- 1.2: CNC job work flow
- 1.3: Visualizing program execution
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- 1.7: 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: Tool length compensation
- 4.3: Cutter radius compensation
- 4.4: Workpiece coordinate system offsets

**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: Hole-machining canned cycles
- 6.2: Working with subprograms
- 6.3: Other special programming features
- 6.4: Programming rotary devices

**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 24 lessons.

Key Concept number seven is two-lesson key concept.

**Points made during the introduction to Key Concept number seven**

Key Concept number seven formally begins the setup and operation part of the course. However, we'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 eighteen 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 certain programming functions to setup people and operators.

While we didn’t go into programming details for setup people and operators, we explained enough to help them understand the setup- and operation-related implications of these programming features. During Key Concepts one and four, for example, we explained enough about program zero assignment, tool length compensation, and cutter 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. We 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 machining 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 machining 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 machining 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 fixture offsets, filled-in offset table, workholding device on the table, cutting tools in the automatic tool changer magazine, 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

We have no suggestions for this lesson.

**To complete lesson 7.1, students must:**

- view the on-line presentation for this lesson (7 minutes)
- read the reading materials for this lesson (30 minutes)

**Exercises**

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

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

**Notes:**

## *Buttons and Switches on the Operation Panels*

**Lesson Plan**

**7.2**

*Explains all of the buttons and switches on a typical CNC machining 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 cove 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 machining 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 machining 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 automatic tool changer or pallet changer).

**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.
- 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 slide show or student manual. 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 can be presented 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.

**To complete lesson 7.2, students must:**

- view the on-line presentation for this lesson (22 minutes)
- read the reading materials for this lesson (30 minutes)

**Exercises**

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

Approximate total study time for this lesson: **62 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. As you can see, there are ten Key Concepts further divided into 24 lessons.

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 machining 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 machining 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 machining 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.*

**1: Know Your Machine from a Programmer's viewpoint**
1.1: Machine Configurations
1.1a: Levil Machining Center Certification Cart
1.2: CNC job work flow
1.3: Visualizing program execution
1.4: Understanding the workpiece coordinate system
1.5: Determining workpiece coordinate system offsets
1.6: Setting workpiece coordinate system offsets
1.7: 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: Tool length compensation
4.3: Cutter radius compensation
4.4: Workpiece coordinate system offsets
**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: Hole-machining canned cycles
6.2: Working with subprograms
6.3: Other special programming features
6.4: Programming rotary devices
**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 24 lessons.

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 several (rather generic) sets of procedures for current controls in the Fanuc product line. You can use any one of them 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 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 8 (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 machining 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 machining 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 tool length measuring), 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)

Be sure to demonstrate the use of the procedures you have developed that are not shown in this lesson.

### Lab exercise

Have students develop a few more 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 (29 minutes)<br>▪ read the reading materials for this lesson (10 minutes) | ▪ Have students take the on-line test for this lesson. (10 minutes)<br>Approximate total study time for this lesson: **49 minutes** |

**Notes:**

Key Concept

**10**

## *Lesson Plan: You Must Know How to Safely Verify Programs*

*Introduces Key Concept number ten.*

**1: Know Your Machine from a Programmer's viewpoint**
- 1.1: Machine Configurations
- 1.1a: Levil Machining Center Certification Cart
- 1.2: CNC job work flow
- 1.3: Visualizing program execution
- 1.4: Understanding the workpiece coordinate system
- 1.5: Determining workpiece coordinate system offsets
- 1.6: Setting workpiece coordinate system offsets
- 1.7: 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: Tool length compensation
- 4.3: Cutter radius compensation
- 4.4: Workpiece coordinate system offsets

**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: Hole-machining canned cycles
- 6.2: Working with subprograms
- 6.3: Other special programming features
- 6.4: Programming rotary devices

**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 24 lessons.

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 machining 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 (24 minutes)
- read the reading materials for this lesson (18 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 (62 minutes)
- 1.1a: Levil Machining Center Certification Cart (15 minutes)
- 1.2: CNC job work flow (37 minutes)
- 1.3: Visualizing program execution (39 minutes)
- 1.4: Understanding the workpiece coordinate system (60 minutes)
- 1.5: Determining workpiece coordinate system offsets (72 minutes)
- 1.6: Setting workpiece coordinate system offsets (47 minutes)
- 1.7: Introduction to programming words (58 minutes)

**2: You Must Prepare to Write Programs**

- 2.1: Preparation for programming (78 minutes)

**3: Understand the motion types**

- 3.1: Programming the three basic motion types (136 minutes)

**4: Know the compensation types**

- 4.1: Introduction to compensation (81 minutes)
- 4.2: Tool length compensation (100 minutes)
- 4.3: Cutter radius compensation (121 minutes)
- 4.4: Workpiece coordinate system offsets (85 minutes)

**5: You must provide structure to your CNC programs**

- 5.1: Introduction to program structure (77 minutes)
- 5.2: Structured program format (87 minutes)

**6: Special features that help with programming**

- 6.1: Hole-machining canned cycles (98 minutes)
- 6.2: Working with subprograms (73 minutes)
- 6.3: Other special programming features (92 minutes)
- 6.4: Programming rotary devices (77 minutes)

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

- 7.1: Tasks related to setup and running production (47 minutes)
- 7.2: Buttons and switches on the operation panels (62 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 (49 minutes)

**10: You must know how to safely verify programs**

- 10.1: Program verification (52 minutes)

Total time to complete the course: Approximately 29 hours

Notes: