

1

CNC Setup and Operations: Cutter Radius Compensation

Objective

Students will be able to:

- ✓ Develop a program using Cutter Radius Compensation (CRC) for a CNC machining center.
- ✓ Analyze the program for errors using the CNC simulator and revise the program as needed.
- ✓ Operate and setup the CNC machine, this will include tool setups, work offsets, program input into the CNC control, and part prove out.

Orienting Questions

- ✓ What is Cutter Radius Compensation?
- ✓ When is Cutter Radius Compensation used in a program?
- ✓ What are the advantages to using Cutter Radius Compensation?
- ✓ What special rules must be followed when using Cutter Radius Compensation?

Except where otherwise noted, this work was created by Glenn Galloway and is licensed under the Creative Commons Attribution 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

1.1 INTRODUCTION TO CUTTER RADIUS COMPENSATION (CRC)

WITH cad/cam SOFTWARE and machine DEMONSTRATION

In this section, **Cutter Radius Compensations** also known as **CRC** will be introduced and illustrated with CAD/CAM software simulator. The use of **CRC** will allow point values to be taken from the drawing without manually compensating for a tool radius. The use of **CRC** will allow various size tools to be used and still maintain the same size part shape.

1.1.1 UNDERSTANDING THE FUNCTION OF CRC

CRC, is a function of the CNC (computer numerical control) controller to automatically shift the cutter edge to the correct position along a cutter path. If it were not for **CRC** the cutter locations would have to be calculated to the cutter center line.

1.1.2 GRAPHIC DEMONSTRATION WHEN NOT USING CRC

Figure 1 will be used to illustrate programming with and without **CRC**.

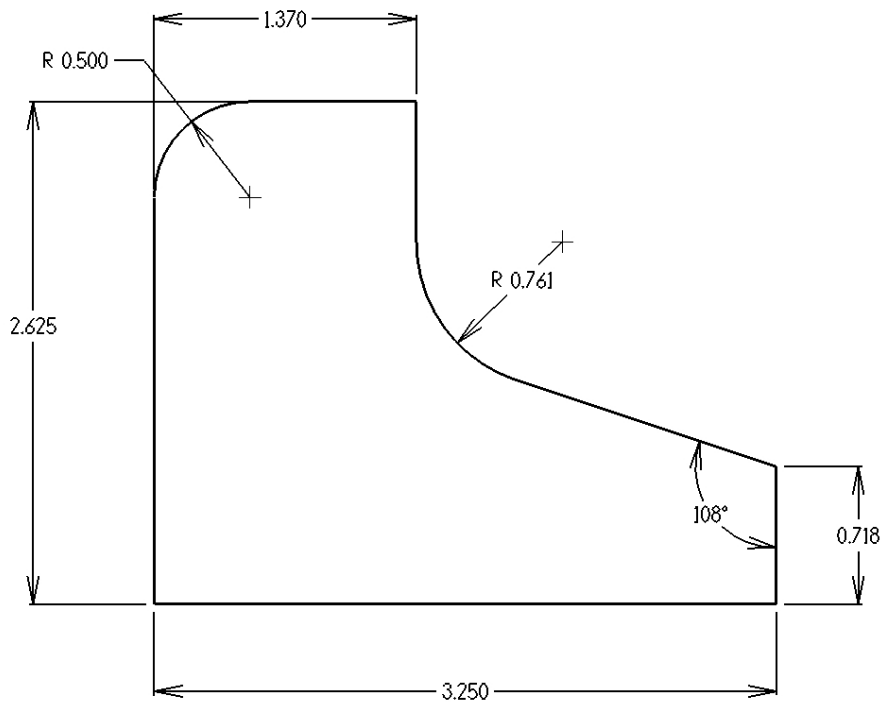


Figure 1, example part print

When programming **without CRC** the tool center locations (X and Y axis) must be programmed. See **figure 2** showing the cutter locations along the tool path that must be programmed to the tool center. One problem with this type of programming is that the tool center location will only make a good part if the tool diameter is always the same size.

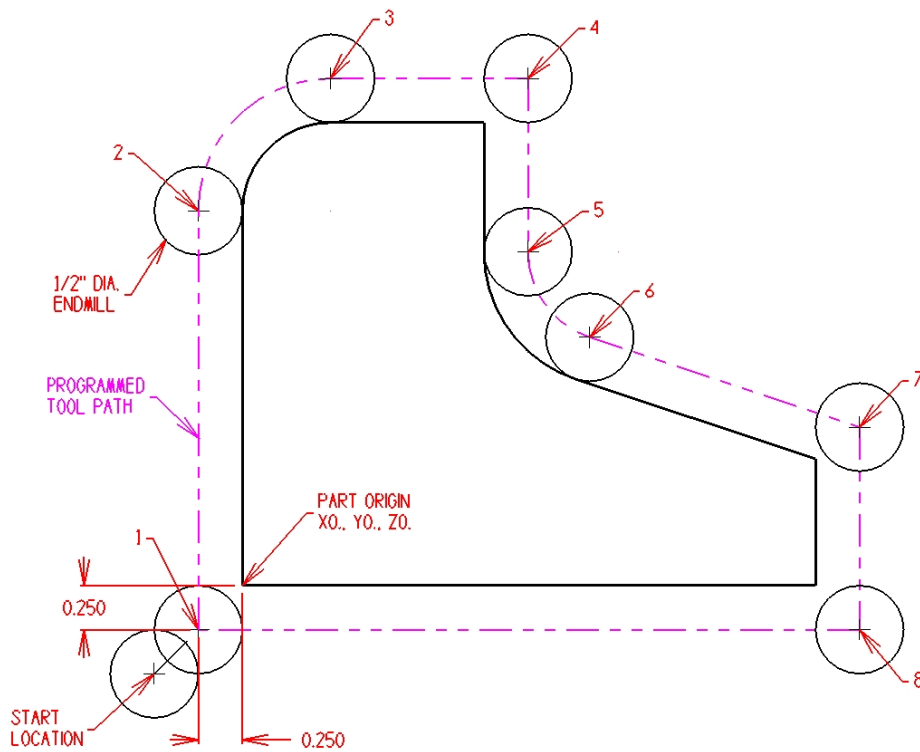


Figure 2, this drawing show the necessary tool path.

Video 1 [shows an animated tool moving around the part shape without CRC active.](#) Remember the points used for programming are offset from the part shape by the amount equal to the radius of the cutter. (Ctrl key + click link to play):

1.1.3 GRAPHIC DEMONSTRATION USING CRC

When programming **with CRC** there is no need to program the tool center, the tool path can be programmed from the actual shape of the part. Tool locations (X and Y Axis) are calculated from the part shape not the tool center. This will allow the use of other size tools to cut the part and will still maintain accurate size. See **figure 3** showing the cutter locations along the tool path that are programmed to the part shape. Notice that CRC keeps the tool tangent to the part shape as it moves from one location to the next.

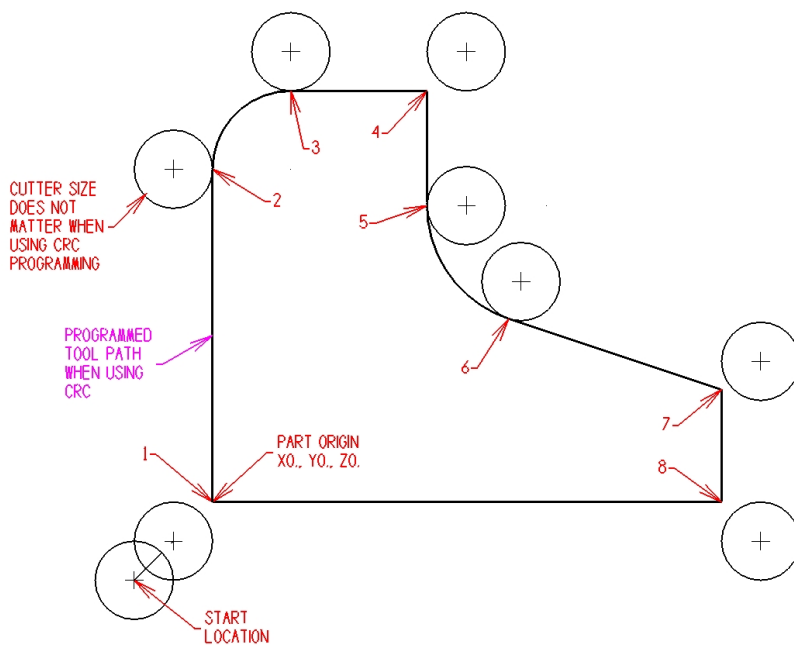


Figure 3 displays the programmed points labeled 1 through 8, and the position of the tool using CRC.

Video 2 [shows an animated tool moving around the part shape with CRC active.](#) Remember the points used for programming are the points along the part shape. (Ctrl key + click link to play):

Using the previous example where programming was done by using the locations along the part shape, see **figure 4** illustration of what the cut would be like if **CRC** was not activated in the program. The result would be a scrap part, the part would be undersize.

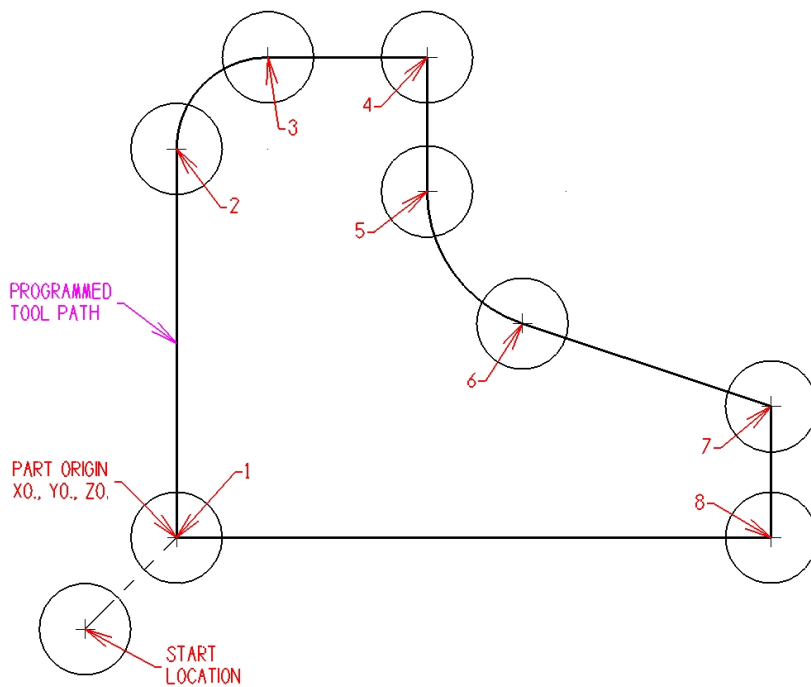
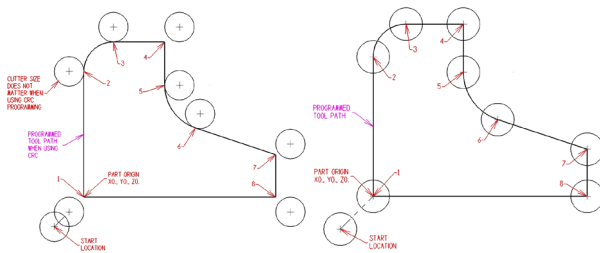


Figure 4

Video 3 [link](#) shows a simulation of the tool moving around the part shape **without CRC active**. Remember the points used for programming are the points along the part shape. (Ctrl key + click link to play):

LABELING ASSESSMENT:

Label the figures below as either "Active" or "Scrap" in the box below.

**1.1.4 MACHINE DEMONSTRATION RESULTS WITH AND WITHOUT CRC**

Video 4 [link](#) was filmed at the CNC machine and will show the use of CRC and what can result if not programmed properly. The video will run on the machine using programs from the simulations shown in videos 1 through 3. To play **Video 4** (Ctrl key + click link to play):

1.2. OVERVIEW OF CUTTER RADIUS COMPENSATION (CRC)

In this section, programming codes, rules, and basic concept for using CRC will be presented and illustrated using the CAM simulator.

1.2.1 CODES USED IN CRC PROGRAMMING

There are three G codes used when programming with CRC.

G40 - cutter radius compensation (CRC) cancel

G41 - cutter radius compensation (CRC) to the left of the programmed contour path

G42 - cutter radius compensation (CRC) to the right of the programmed contour path

Compensation to the right or left is determined by the direction the tool will be traveling from one point to the next and which side of the line the tool needs to be on to make the correct cut.

Visualize this: If you could stand at the current tool location and view toward the next location, you can then see if you need the cutter to be on the right or left of the line as it moves from the current location to the next location. See **figure 5**.

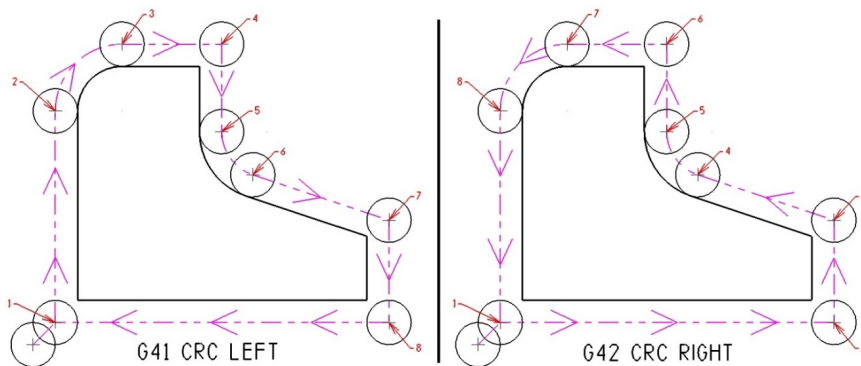


Figure 5

Notice in **figure 5** the tool path direction and the side of the part contour line that the tool is on as it moves from a point to the next point. When viewing from a point to the next point the cutter left (G41) or cutter right (G42) is determined by the side of the line the cutter needs to be on as it cuts the part contour shape.

1.2.2 ADDITIONAL CODES USED IN CRC

When the code G41 or G42 are programmed it is necessary to add an additional program word to the block. The CNC control will need information about the radius of the tool that is to be compensated.

Let's review for a moment of some information you should already know. From the tool offset screen there is the "GEOM (H)" column which holds the length value for the tool. You should keep in mind that when programming the first move to approach the part in the Z axis, programming must be two words, G43 H##, for example if you are using tool "05" in the spindle then it would be G43 H05 Z1.0. The number "05" is to match the number of the tool in the spindle.

Looking further at the tool offset screen you will find another column that is the "GEOM (D)" column that needs to have the radius value of the tool. Note: On various FANUC models the tool offset screen may not look the same. Some have the tool "WEAR" columns beside the "H" tool length and "D" tool radius columns. See **figure 6**.

"Tool Offset" page:

OFFSET					O0000	N00000
NO.	GEOM (H)	WEAR (H)	GEOM (D)	WEAR (D)		
001	3.9880	0.0000	1.5000	0.0000		
002	3.8360	0.0000	0.0625	0.0000		
003	5.4460	0.0000	0.2500	0.0000		
004	5.2160	0.0000	0.6250	0.0000		
005	4.0960	0.0000	0.3750	0.0000		
006	15.0000	0.0000	0.0000	0.0000		
007	15.0000	0.0000	0.0000	0.0000		
008	15.0000	0.0000	0.0000	0.0000		
RELATIVE	X	0.0000	Y	0.0000		
	Z	0.0000	IA	0.000		

Figure 6

The programming of **G41** and/or **G42** must contain a "**D##**" word so the control will know where to find the radius value to compensate for. Programming for a block to turn on **CRC** for "tool 05" will be as follows.

G01 **G41 D05** X(value) Y(value)

D05 was used because tool 05 is in the spindle. So for review, if you are using Tool "05" in the spindle then the length offset would be G43 H05 and CRC compensation would be G41 or G42 D05.

There are some other programming examples used where other offset numbers can be used with a particular tool number. This is sometimes used to create rough cuts by using false values in the offset screen to cause the machine to increase or decrease the size of the part without changing the program values. For example: I have a machine that can hold up to 24 tools. I would save the "H" and "D" values for the first 24 positions of the tool offset screen for the actual true size of the tool. For this example we will be using tool 05.

If programming finish work, use G43 H05 for the length and G41/G42 D05 for the radius offset. If programming roughing work use numbers that are above 24 and up to 99. These are offsets that normally are never used because the machine in this example will only hold 24 tools. For example: To program a roughing pass, first pick a number that is not being used, I will use number 55, and type in a value in the offset for 55 that is larger than the radius for the tool we are using by the amount that you desire to leave

on the part. Call the offset in the program just as before but replace the offset number with the number of the false value.

G01 G41 D55 X(value) Y(value) this will turn on **CRC** to the left and offset the tool by the amount in offset number 55. When you are ready to program the finish cut with the same tool, program G01 G41 D05 X(value) Y(value). Offset "05" will have the actual radius size of the tool, this will remove the remaining material left on the part from using offset "55".

ACTIVITY

Let's use the same example listed above where a machine that can hold up to 24 tools is used, the "H" and "D" values for the first 24 positions of the tool offset screen for the actual true size of the tool were saved, and we are using tool 05. Match the different codes to the function. Then, click submit.

- | | |
|---------------------------------|--|
| G01 G41 D55 X(value) Y(value) (| Use when programming finish work for the length |
| G01 G41 D05 X(value) Y(value) | Turns on CRC to the left and offset the tool by the amount in offset number 55 |
| G01 G41 D05 X(value) Y(value) | Use when ready to program the finish cut with the same tool |

G41/G42 D05 Use when ready to program the finish cut with the same tool Use when programming finish work for the radius offset

ANSWER

- | | |
|---------------------------------|--|
| G01 G41 D55 X(value) Y(value) (| Turns on CRC to the left and offset the tool by the amount in offset number 55 |
| G01 G41 D05 X(value) Y(value) | Use when ready to program the finish cut with the same tool |
| G01 G41 D05 X(value) Y(value) | Use when programming finish work for the length |

G41/G42 D05 Use when programming finish work for the radius offset

1.2.3 PROGRAMMING RULES WHEN USING CRC

There are a few special rules that must be applied to the programming method when using CRC:

1. The approach to the desired contour cut and the exit from the contour cut is critical for compensation to work correctly. The approach distance must be equal to or greater than the radius of the tool being compensated. **The first motion block used to start compensation** called “ramping on move” must be along a linear path (G01 or G00), arc moves (G02, G03) are only allowed after the compensation is started. The same rule applies on **the last block that is used to turn off compensation** called “ramping off move”.
2. CRC can only be applied to 2 axis of motion at a time even though there is 3 or more axis available for a particular machine. The plane available for compensation will be determined by the plane selection called out in the program, G17 (XY plane), G18 (XZ plane), or G19 (YZ plane).
3. If programming in G17 (XY plane), an example of a non-motion block would be a Z axis move or a block with no axis motion like M-code blocks etc., or any blocks that do not contain an X, Y, or X and Y. The use of more than one non motion block in a row is not allowed. More than one non motion block in a row interrupts the control from calculating proper cutter location for the next move. An alarm will be issued at the control if this programming error is encountered.

1.3 BASIC PROGRAMMING USING CUTTER RADIUS COMPENSATION

In this section, program planning, point values solved, and program writing.

1.3.1 STUDY THE PRINT AND PLAN MACHINING OPERATIONS

To prepare for the exercise in 1.3.3, let's look at the following figure. For this example we will program a contour operation to cut around the shape of the part to illustrate how CRC is applied in the program. See **Figure 7 Sample Print 1**.

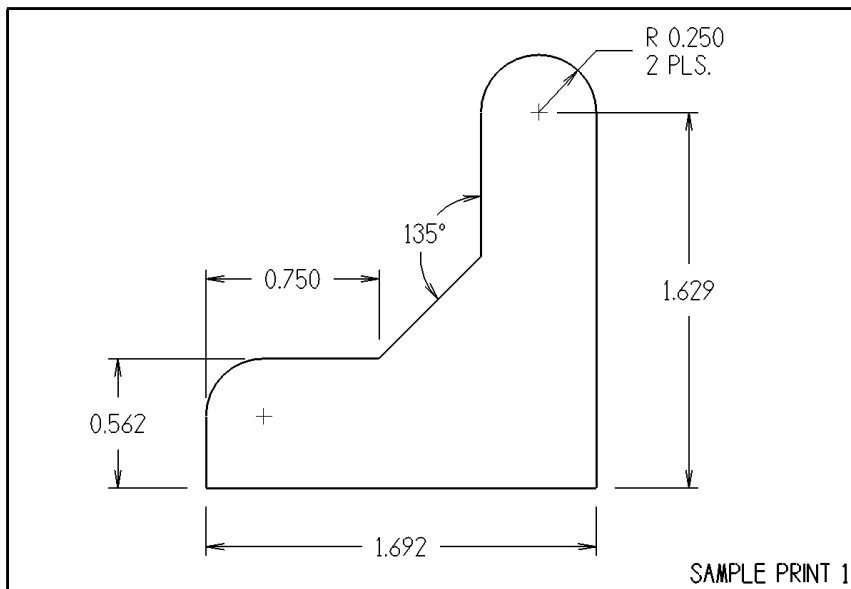


Figure 7. Sample Print 1

1.3.2 POINT IDENTIFICATION AND SOLVE X AND Y VALUES TO USE IN PROGRAMMING

Now let us breakdown it down. There are several steps involved in the programming process as follows:

1. Determine where is the best place on the shape for the part origin. This will be where all points along the drawing are calculated from. See **figure 8** for the origin location that will be used.

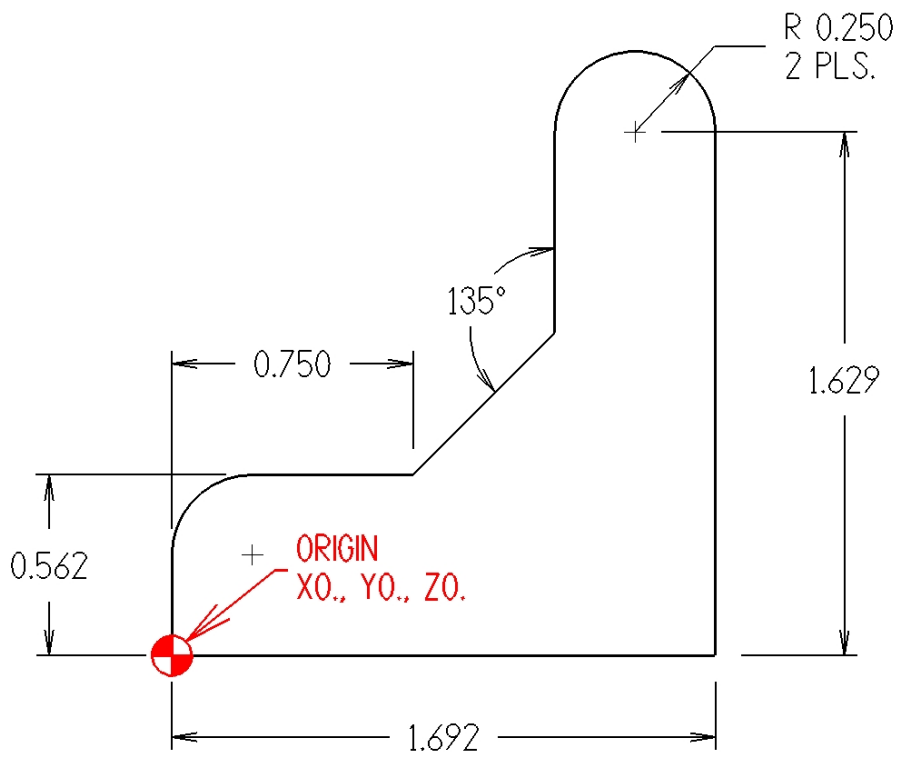


Figure 8, origin placement on the drawing

2. Place a dot and a number next to the dot for each programming tool position as shown in figure 9.

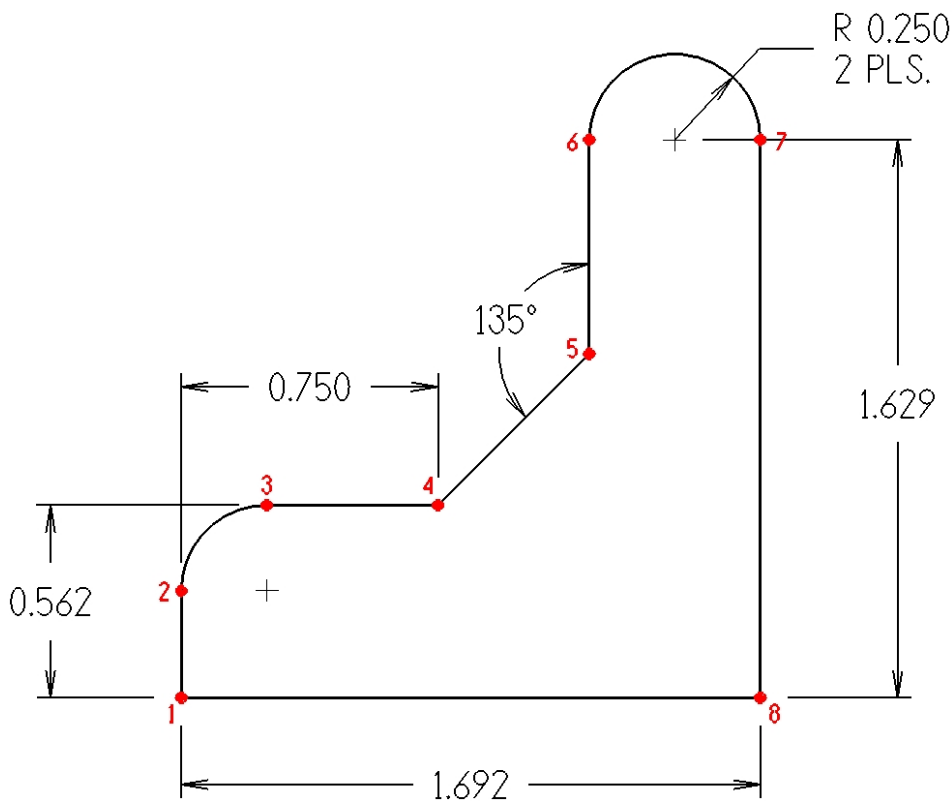


Figure 9, points are placed on the programming points needed, and numbered to match the point chart, these will be used for programming.

1.3.3 POINT SOLVING

ACTIVITY 1,

Solve the points shown in figures 7 through 9 and fill in the values in the chart below.

Solve the points in figure 9 and fill in the point values in the chart below.

Mill Points for Project: Sample Print 1					
Point #	X Value	Y Value	R Value	I Value	J Value
1					
2					
3					
4					
5					
6					
7					
8					

Answers may be found in the assessment guide at the end of this module.

1.3.4 WRITING THE PROGRAM USING CRC

Notice the codes in the program, G41 D01 and G40, for proper tool ramp on/off moves for CRC.

%

O1234 (SAMPLE PART USING CRC);

N10 G17 G20 G40 G80 (SAFE BLOCK);

N15 T01 M06 (PUT TOOL INTO THE SPINDLE);

N20 G00 G90 G54 X-.75 Y-.75 S1200 M03 (START SPINDLE AND TRAVEL TO A LOCATION OFF PART WHERE THE Z AXIS CAN BE MOVED DOWN ON THE NEXT BLOCK);
N25 G43 H01 Z1.0 (TOOL LENGTH OFFSET ACTIVE, TRAVEL Z AXIS TO 1.0 INCH ABOVE PART)
N30 Z.1 (MOVE CLOSER TO PART SURFACE);
N35 G01 Z-.125 F15.0 (FEED THE TOOL TIP BELOW THE PART SURFACE);
N40 G41 D01 X0. Y0. (THIS IS KNOWN AS THE RAMP ON MOVE, TURNS ON CRC TO THE LEFT AS THE TOOL MOVES TO POINT 1 FROM OUR POINT CHART);
N45 Y.312; (CUT TO POINT 2);
N50 G02 X.25 Y.562 R.25 (CUT CLOCKWISE ARC TO POINT 3);
N55 G01 X.75 (CUT TO POINT 4);
N60 X1.192 Y1.004 (CUT TO POINT 5);
N65 Y1.629 (CUT TO POINT 6);
N70 G02 X1.692 Y1.629 R.25 (CUT CLOCKWISE ARC TO POINT 7);
N75 G01 Y0. (CUT TO POINT 8);
N80 X0. (CUT BACK TO POINT 1 TO COMPLETE THE FULL CONTOUR);
N85 G40 X-.75 Y-.75 (THIS IS KNOWN AS THE RAMP OFF MOVE, TURNS OFF CRC);
N90 G0 Z1.0 (RAPID Z AXIS UP OFF PART);
N95 G91 G28 Z0. M5 (RETRACT Z TO MACHINE HOME AND STOP THE SPINDLE);
N100 G28 Y0. (MOVE THE TABLE TO THE FRONT OF THE MACHINE FOR UNLOADING THE PART);
N105 M30 (PROGRAM END);
%

SUMMARY

In this section a summary of Cutter Radius Compensation (**CRC**) will be looked at, its use, and advantages.

Cutter Radius Compensation also known as **CRC**, is a function of the CNC controller to be able to automatically shift the cutter edge to the correct position along a cutter path. If it were not for CRC the cutter locations would have to be calculated to the cutter center line.

Cutter Radius Compensation (CRC) is developed and used to program continuous path milling operations. Programming with CRC should be used on most contour shapes, this allows other tools to be used if needed without changing the part shape.

There are three main advantages to using **Cutter Radius Compensation (CRC)** programming.

1. The programmer can use point values that are directly along the cutter path rather than having to calculate the tool center.
2. During production runs, cutters of different sizes can be used such as undersize tools when the original size tool is no longer available. It is a simple radius offset value change in the machine control to achieve this. The programming will remain the same.
3. A program section or operation can be used over and over with different tools to create roughing passes and finish passes. This is done by changing the radius offset value and using multiple offsets.

There are a few special rules that must be applied to the programming method when using **CRC**.

1. The point before the **CRC** approach and exit from the contour cut must be equal to or greater than the cutter radius.
2. **CRC** can only be started and stopped along linear motion.
3. **CRC** can only be applied to 2 axis of motion at a time.
4. The use of more than one non motion block in a row is not allowed.

KEY TERMS

CRC Abbreviation for Cutter Radius Compensation, this is used in programming to offset the cutter from the programmed path by the amount equal to the tool radius. There are special applications that the value may be larger or smaller than the radius of the tool to create rough cuts etc.

G40 CRC Cancel

G41	CRC to the Left
G42	CRC to the Right
D##	When programming G41/G42 the CNC controller needs to know how much to compensate the tool radius. The "D word" with the "##" must be used when activating CRC. The value is held in the controller, in the offset section, the ## usually will match the current tool number being used.
Non-Motion Block	This is a block where no motion in the two axes being compensated. Example: compensation can only be in two axes at once, so a block that has a 3 rd axis movement would be a non-motion block in the compensated plane. There cannot be 2 non-motion blocks in a row or it will cause CRC to fail.
Ramp on/off	This is a program block where CRC is started or stopped, tool motion in the two axes to be compensated must be programmed to travel a minimum of 1/2 the diameter of the end mill to allow compensation to be started or stopped fully.

LABS

LAB 1. PHYSICAL LAB USING A 3 AXIS MACHINING CENTER

- Lab exercise will be done at the CNC machine using a 3 axis machining center, or if not available a CNC simulator can be used.

-Materials needed are the CNC program, 4" x 4" wax block, and 1/2 diameter end mill. Tools will be loaded into the machine and tool lengths set, program entered into the CNC machine, wax block loaded into the machine vise, and the part origin set (G54).

-Requirements for successful lab completion will be to program, setup and prove out a CNC program using **CRC**.

ASSESSMENT

DISCUSSION QUESTIONS

1. In industry when writing a program, what are the advantages of using **CRC** over not using **CRC** in programming contour tool paths?
 2. Discuss how using **CRC** in programming will allow the use of other cutter sizes without changing the original program, and why is this useful in industry.
2. _____

Commented [LEH1]: Using CRC in programming contoured tool paths allows the cutter to be shifted toward or away from the part to remove or leave more material depending on the application. This allows for an adjustment in size to closely maintain tolerances and produce quality parts. If CRC is not used then the programmer/operator is at the mercy of the diameter of the tool and having zero wear on the cutting properties of the tool.

Commented [LEH2]: Using CRC will allow cutter sizes to be changed as long as the correct information is input into the controller for that tool. Making sure that the radius of the tool matches with what is being used to cut the material. No changes to the program will need to be made because the tool path was programmed using the center of the tool. In industry this is beneficial in saving time programming but is limited to what diameter tools are available to the machinist.

Formatted: Indent: Left: 0.5", No bullets or numbering

MULTIPLE CHOICE, TEST 1

Instructions: Select the best answer.

1. When programming with **CRC (cutter Radius Compensation)** along a contour cut, coordinates are calculated
 - A. from the tool centerline to the part origin.
 - B. from the tool tip.
 - C. from the corner of the machine table.
 - D. from the exact part shape to the part origin.
2. When programming with **CRC**, this will allow
 - A. All three apply
 - B. cutters of different sizes to be used and still maintain part tolerance.
 - C. programming the part shape to actual size.
 - D. rough and finish passes can be made with the same program section using different radius values in the offset.
3. **CRC** can only be started and stopped
 - A. by lowering and raising the Z axis.
 - B. on a linear motion block.
 - C. on an arc motion block.
 - D. when a circle is programmed on the same block.
4. How many axes can **CRC** control at once?
 - A. 2
 - B. 3
 - C. 4

- D. 1
5. How many non-motion blocks in a row can be programmed without causing compensation to fail?
- A. 1
B. 3
C. 2
D. None
6. What other code must be programmed on the "ramp on" block to start **CRC** active?
- A. "R" code
B. "H" code
C. "T" code
D. "D" code
7. What value is entered in the offset screen for tools that **CRC** will be programmed?
- A. The tool length.
B. The tool radius.
C. The tip shape.
D. The tool diameter.
8. When programming a contour cut around the outside of a shape, traveling clockwise around the part, which compensation side (which G code) will need to be programmed?
- A. G41
B. G40
C. G43
D. G42
9. **CRC** can be used to make what type of cuts?
- A. 3D contouring.
B. Inside and outside contour cuts.
C. None of these choices
D. Drilling operations.

Continued

10. To make a rough cut with a 1/2 inch diameter end mill, that will leave .03 of stock on the surface for finishing later, with the same programming coordinates, what should be the value to enter in the offset screen?
- A. .2200
B. .2800
C. .5300
D. .4700

SC ACCELERATE

MTT 252

Open Text

CNC Setup and
Operations (Hybrid)

Module#1

Cutter Radius
Compensation

SC ACCELERATE

MTT 252

Open Text

CNC Setup and
Operations (Hybrid)

Module#1

Cutter Radius
Compensation

FURTHER STUDY

OTHER SUPPORTING INFORMATION

Internet search for CNC machining with **CRC** in www.Youtube.com will show some videos. Internet search in www.Google.com for CNC using CRC will find some reading material.

CRITICAL THINKING

Compare and contrast key concepts programming with CRC and programming without CRC by completing the table below.

	Programming without CRC	Programming with CRC
Define or explain each concept		
Explain how the concepts are similar		
Explain how each concept is different with respect to specific attributes		

2. When identifying points and solving x and y values, there are two steps involved when CRC is applied to the programming process of a contour operation. Describe each step in sequence.

ANSWERS TO ASSESSMENTS**LABELING ASSESSMENT ANSWER**

The first illustration is active, the second is scrap.

ANSWERS TO TEST 1

1. D
2. A
3. B
4. A
5. A
6. D
7. B
8. A
9. B
10. B

ANSWERS TO ACTIVITY, POINT SOLVING

Answers: Mill Points for Project: Sample Print 1

Point #	X Value	Y Value	R Value	I Value	J Value
1	0	0			
2	0	0.312			
3	0.25	0.562	0.25		
4	0.75	0.562			
5	1.192	1.004			
6	1.192	1.629			
7	1.692	1.629	0.25		
8	1.692	0			

DISCUSSION QUESTION ANSWERS

Using CRC in programming contoured tool paths allows the cutter to be shifted toward or away from the part to remove or leave more material depending on the application. This allows for an adjustment in size to closely maintain tolerances and produce quality parts. If CRC is not used then the programmer/operator is at the mercy of the diameter of the tool and having zero wear on the cutting properties of the tool.

Using CRC will allow cutter sizes to be changed as long as the correct information is input into the controller for that tool. Making sure that the radius of the tool matches with what is being used to cut the material. No changes to the program will need to be made because the tool path was programmed using the center of the tool. In industry this is beneficial in saving time programming but is limited to what diameter tools are available to the machinist.

ANSWERS TO CRITICAL THINKING QUESTIONS

1. Compare and contrast key concepts programming with CRC and programming without CRC by completing the table below.

	Programming without CRC	Programming with CRC
Define or explain each concept	When programming without CRC the tool center locations (X and Y axis) must be programmed.	Program with CRC is used in continuous path milling operations and should be used on most contour shapes, this allows other tools to be used if needed without changing the part shape.
Explain how the concepts are similar	When the program positions are manually calculated, the part will be the correct size. When program pieces are calculated directly from the part shape and CRC is used, the part will be the correct size.	

	Programming without CRC	Programming with CRC
Explain how each concept is different with respect to specific attributes	To make the part the correct size, the cutter size must always be the exact size that was used to calculate the program positions.	Using CRC, the cutter does not have to remain the same size, other cutter sizes can easily be used by adjusting the offset that controls the shift amount to cut the part to the correct size.

2. When identifying points and solving x and y values, there are two steps involved when CRC is applied to the programming process of a contour operation. Describe each step in sequence.

ANSWER

Step 1. Determine where is the best place on the shape for the part origin. This will be where all points along the drawing are calculated from.

Step 2. Place a dot and a number next to the dot for each programming tool position.

ATTRIBUTION TABLE

Author/s	Title	Source	License
Glenn Galloway	Video 1 through 4	Glenn Galloway YouTube.com	Creative Commons Attribution 3.0 Unported License
Glenn Galloway	Figure 1 through 9	Glenn Galloway	Creative Commons Attribution 3.0 Unported License



This workforce solution was funded by a grant awarded by the U.S. Department of Labor's Employment and Training Administration. The solution was created by the grantee and does not necessarily reflect the official position of the U.S. Department of Labor. The Department of Labor makes no guarantees, warranties, or assurances of any kind, express or implied, with respect to such information, including any information on linked sites, and including, but not limited to accuracy of the information or its completeness, timeliness, usefulness, adequacy, continued availability or ownership.