Special Codes

Table of Contents

Introduction to Special Codes: 2

1 Special Characters 4

2 Joining points with lines/polylines (Example 1 & 2) 4

3 Special Codes 5

3.1 "/": Append Description (Example 3) 5

3.2 "///": Replace Description (Example 4) 6

3.3 "": Prefix Description (Example 5) 6

3.4 "//": Append Field Code Description 6

3.5 "\": Prefix Field Code Description 7

3.6 "|": End Coding 7

3.7 MULT: Multiple Field Code 7

3.8 PC: Start Curve (Example 6) 7

3.9 PT: End Curve (Example 7) 8

3.10 AFIT: Fit Arc (Example 8) 9

3.11 BFIT: Best-Fit Line (Example 9) 10

3.12 CTOG: Curve Toggle 10

3.13 CLO: Close (Example 10) 10

3.14 GAP: Gap (Example 11) 10

3.15 NE: No Elevation (Example 12) 11

3.16 YZ: Yes Elevation (Example 13) 11

3.17 NOS: Non-Surface 11

3.18 ZO: Elevation Only 11

3.19 PHOTO 12

3.20 LABEL: 12

3.21 COLOUR 12

3.22 Offsets: OH, OV, OFL (Example 14a & 14b) 13

3.22.1 Multiple Offsets (Example 15) 14

3.22.2 Offset both Left & Right (Example 16) 14

3.23 Symbol Size (Example 17) 15

3.24 ROT: Rotate (Example 18) 16

3.25 SMO: Smooth (Example 19) 16

3.26 AZI & DIST 17

3.27 JOG: Extend by Distance (Example 20) 17

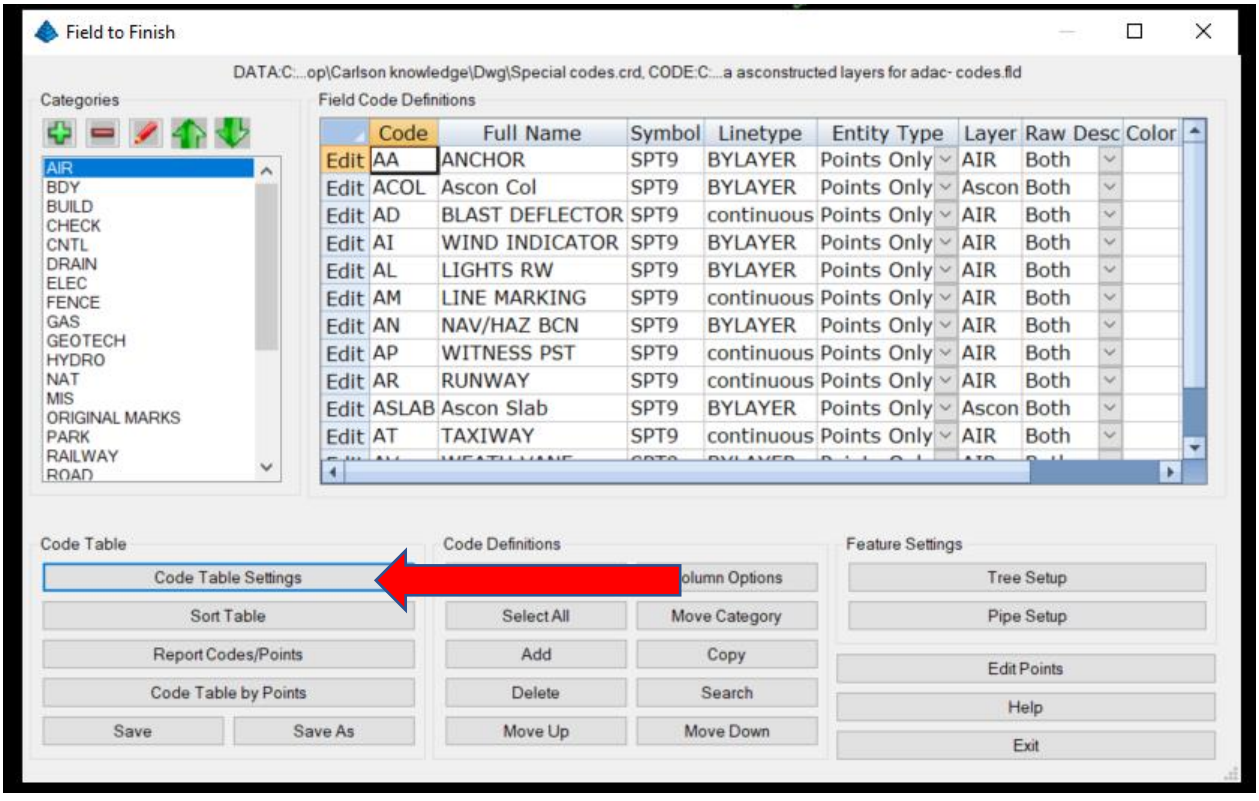
3.28 JPN: Join to Point Name (Example 21) 17

3.29 XSCT: Template (Example 22) 18
3.30 NEAR: Nearest Found 18
3.31 RECT: Close Rectangular (Example 23a &23b) 18
3.32 LTF: Lintype Flip (Example 24) 19
3.33 LTW: Line Width (Example 25) 19
3.34 PARKING: Parking (Example 26) 20
3.35 RAMP: Kerb Ramp (example27) 20
3.36 CIR: Circle: (Example 28 & 29) 21

Introduction to Special Codes:

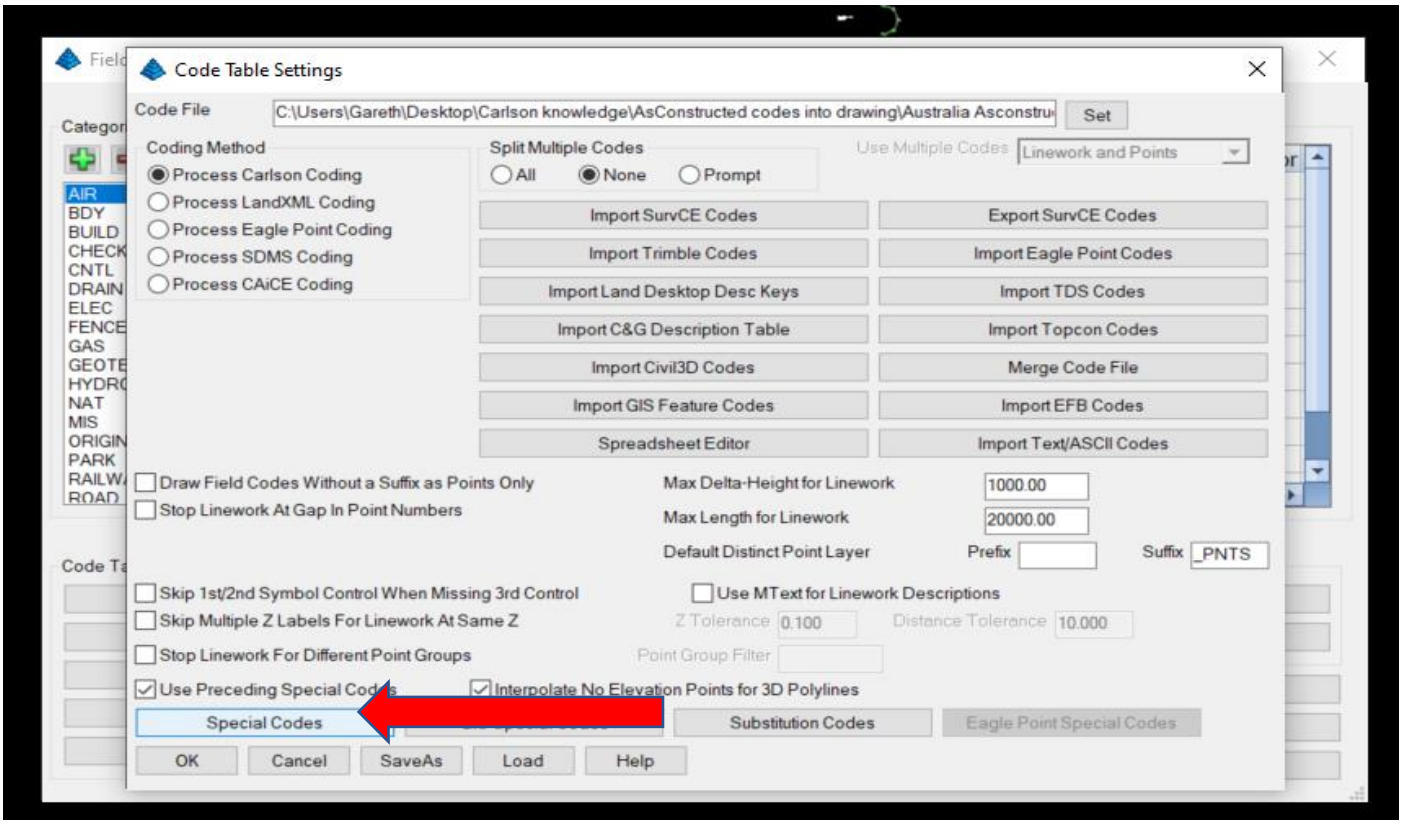
This section allows you to substitute the existing predefined special codes and characters with your own. Draw Field to Finish recognizes several special codes. A special code is placed before or after the regular code with a space separating the code and special code. The Append Desc Auto Space option applies to the special codes that control the point description label. This option sets whether to insert a space in the description label when appending to the description.

Here is a listing of the default special codes and characters. You can find the Special code by selecting 'Code Table Settings'

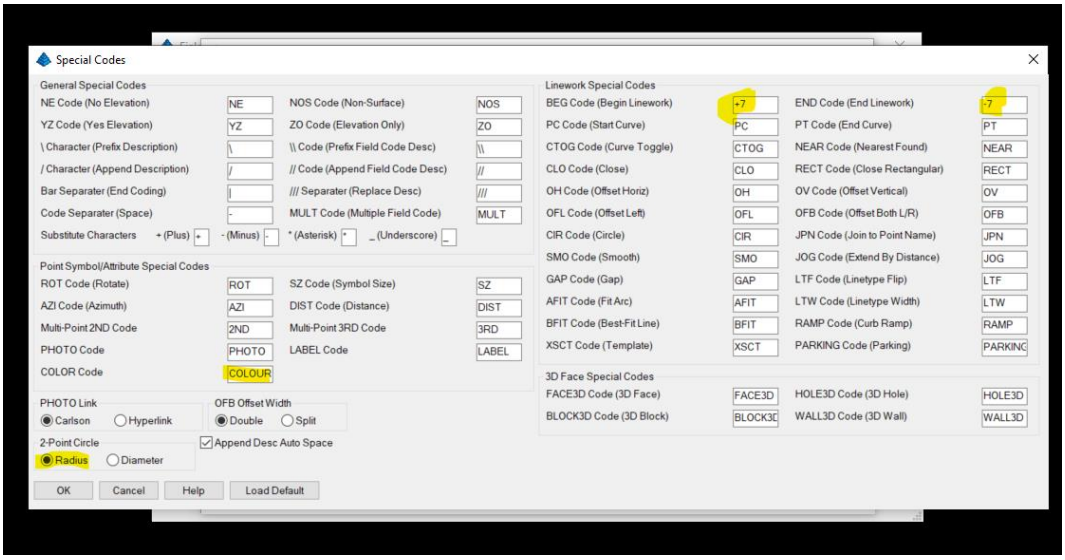


	Code	Full Name	Symbol	Linetype	Entity Type	Layer	Raw Desc	Color
Edit	AA	ANCHOR	SPT9	BYLAYER	Points Only	AIR	Both	
Edit	ACOL	Ascon Col	SPT9	BYLAYER	Points Only	Ascon	Both	
Edit	AD	BLAST DEFLECTOR	SPT9	continuous	Points Only	AIR	Both	
Edit	AI	WIND INDICATOR	SPT9	BYLAYER	Points Only	AIR	Both	
Edit	AL	LIGHTS RW	SPT9	BYLAYER	Points Only	AIR	Both	
Edit	AM	LINE MARKING	SPT9	continuous	Points Only	AIR	Both	
Edit	AN	NAV/HAZ BCN	SPT9	BYLAYER	Points Only	AIR	Both	
Edit	AP	WITNESS PST	SPT9	continuous	Points Only	AIR	Both	
Edit	AR	RUNWAY	SPT9	continuous	Points Only	AIR	Both	
Edit	ASLAB	Ascon Slab	SPT9	BYLAYER	Points Only	Ascon	Both	
Edit	AT	TAXIWAY	SPT9	continuous	Points Only	AIR	Both	

Then selecting 'Special Codes'



Here is a listing of the default special codes and characters. Some codes will need to be changed. (See highlighted)



1 Special Characters

The characters (*, -, +, /, and _) can be used and substituted in Draw Field to Finish. The way these characters are used is that when the file is processed the description field is searched for these characters. If the "+" symbol was changed to "-" then the program would look for "-" and change it to "+". This is useful when a particular data collector may not have all the symbols available. With these substitutions you can make a character that is provided on the data collector generate the symbol needed. Multiple characters can also be used. For example, "--" can be used to in order to produce a "/" character or any of the characters listed above.

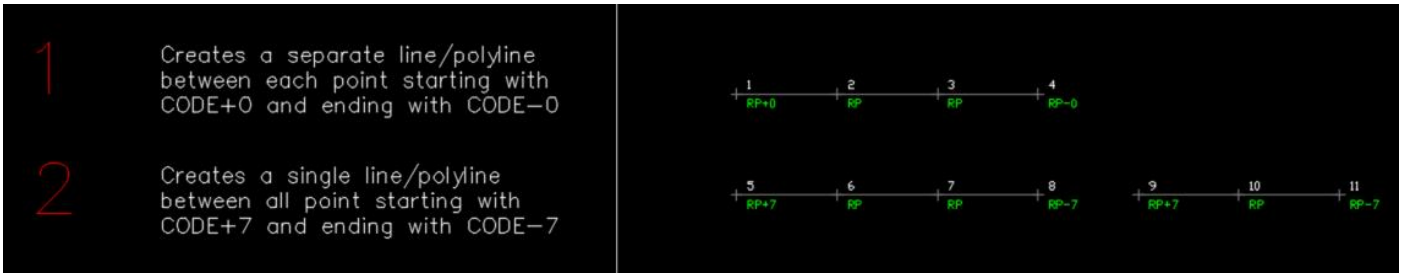
2 Joining points with lines/polylines (Example 1 & 2)

There are two different methods for connecting linework. One method creates line work by connecting points with the same code. The linetype is defined by the code as either points only (no line work), lines, 2D polylines, both 2D and 3D polylines, or 3D polylines (breaklines). Distinct lines with the same code are defined by adding a group number to the end of the code name in the data file. With this method, all points with the description CODE1 will be one line while points with CODE2 will be another line. Both CODE1 and CODE2 use the definition for CODE. For example, the code RP could be a code for edge of road pavement that is to be connected as 3D polylines. If there are two separate edge of pavement lines on the left and right sides of a road, all the points for the left side could have the description RP1 and the points on the right side could be RP2. Besides having the number after the code, the number can be used as a prefix by defining the code with a # special character. For example, when the code is defined as #CODE, then the points with descriptions 10CODE and 20CODE get matched to this code.

The second method is the PointCAD format. This method also connects points with the same code. The difference is that instead of using a number after the code for distinct lines, you use the same code with an additional code for starting and ending the line, a space is used to separate the code and special code. For example, +0 is used to start a line and -0 to end, so the coding for a segment of edge of pavement could be RP +0, RP, RP, RP -0. Another special code that has been added to Field to Finish is +7, -7. These 7 codes will use the linetype definition of line, 2D polyline or 3D polyline defined by the Draw Field to Finish

code. For example, if RP is defined as a 3D polyline, then the coding RP +7, RP, RP, RP -7 will create a 3D polyline. Otherwise, codes like +0, -0, which is defined as start and end line, will draw RP as a line. Other PointCAD special codes are: +4 starts a curved 2D polyline, +5 starts a 3D polyline, +7 starts a line whose type is specified by the field code definition, +8 starts a 2D and 3D polyline combination. //, followed by a field code, concatenates that field code's description on to the point's description. For example, OAK//04 might become LIVE OAK TREE 4m if the field code OAK translates to LIVE OAK TREE and the field code 04 translates to 4m.

The advantage to the PointCAD method is that you don't have to keep track of line numbers. For example, if you are surveying 50 kerb lines, the first method would require you to use 50 distinct kerb numbers. The advantage to the first method is that you don't have to use the start and end codes. Also, the Nearest Found connection option applies to the first method.

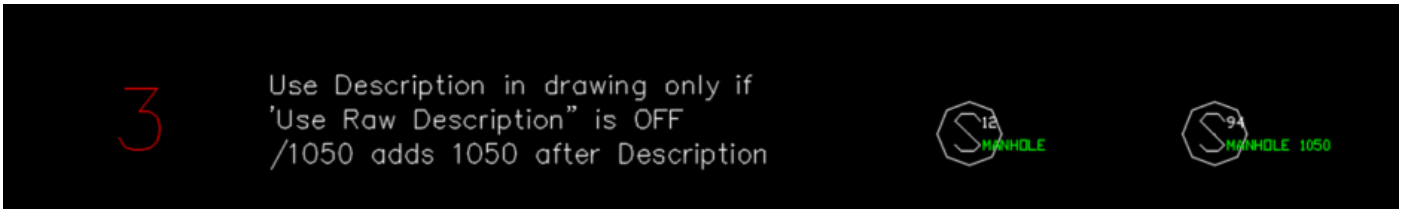


3 Special Codes

3.1 "/": Append Description (Example 3)

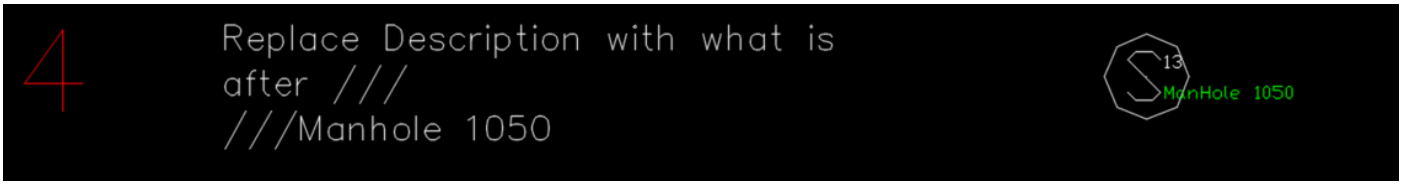
Carlson points in the drawing have point attributes including a description. When Field-to-Finish draws the points, the point description from the coordinate file is processed to match a code. The code then defines the description that is drawn with the point. For example, consider a code of "SM" with a description of "MANHOLE" and a data point with the description "SM". The data point description "SM" would be matched with the code "SM" and the point would end up being drawn with the description "MANHOLE". A special character "/" (the forward slash or divide key) can be used for an unprocessed description to append. Everything after the "/" is added directly to the point description and is not considered a code and no further substitution is done on it. For example, a data point with the description "SM / 1050" with the same

code "SM" definition above would be drawn with the description "MANHOLE 1050".



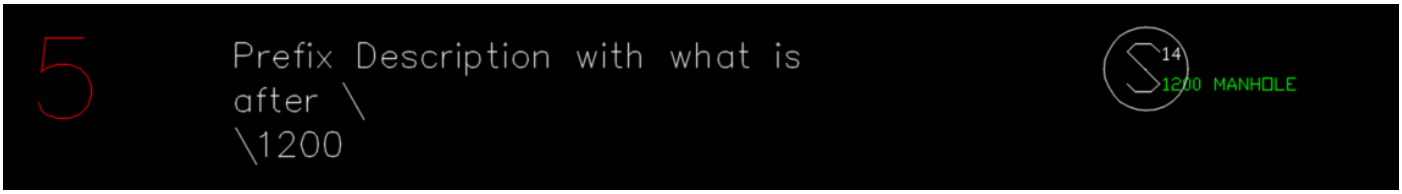
3.2 "///": Replace Description (Example 4)

This special code takes the part of the description after the "///" and uses it as the point description label.



3.3 "\": Prefix Description (Example 5)

This special code takes the part of the description after the "\" and puts it as the prefix before the point description. For example, a data point with the description "SM \ 1200" and a "SM" code definition with a description of "MANHOLE" would be drawn with a description of "120 MANHOLE".



3.4 "//": Append Field Code Description

This special code causes text after the "//" to be interpreted as a field code. That field code's description is then appended to the first field code's description. For example, if the field code 02 has the description *2m* and the field code OAK has the description *oak tree*, then 02//OAK will result in the point having the description of *2m oak tree*. If the "/" character has been replaced with a different character, for example with a & character, then the "//" code would become "&&".

3.5 "\\": Prefix Field Code Description

This special code is the same as "//" except that field code's description is then prefixed instead of appended to the first field code's description.

3.6 "|": End Coding

The bar separator indicates the end of coding. Everything after the bar is ignored for Field-to-Finish processing.

3.7 MULT: Multiple Field Code

This code applies when the Split Multiple Codes under Code Table Settings is set to None and you want to override this setting and explicitly spilt selected codes. Multiple codes apply to points with dual code definitions for drawing two different style points or for connecting different linework to the same point. For example, if a point is both a footpath and driveway corner, then the point description could be "FP MULTDR".

3.8 PC: Start Curve (Example 6)

This code begins a three-point arc or a curved line when used with the "RM" code (see below). The point with this special code is the first point on the arc. The next point with the code is considered a point on the arc, and third point with the code is the arc endpoint. For example (in point number, X, Y, Z, description format),

15, 2000, 1994, 0, RM PC - start curve

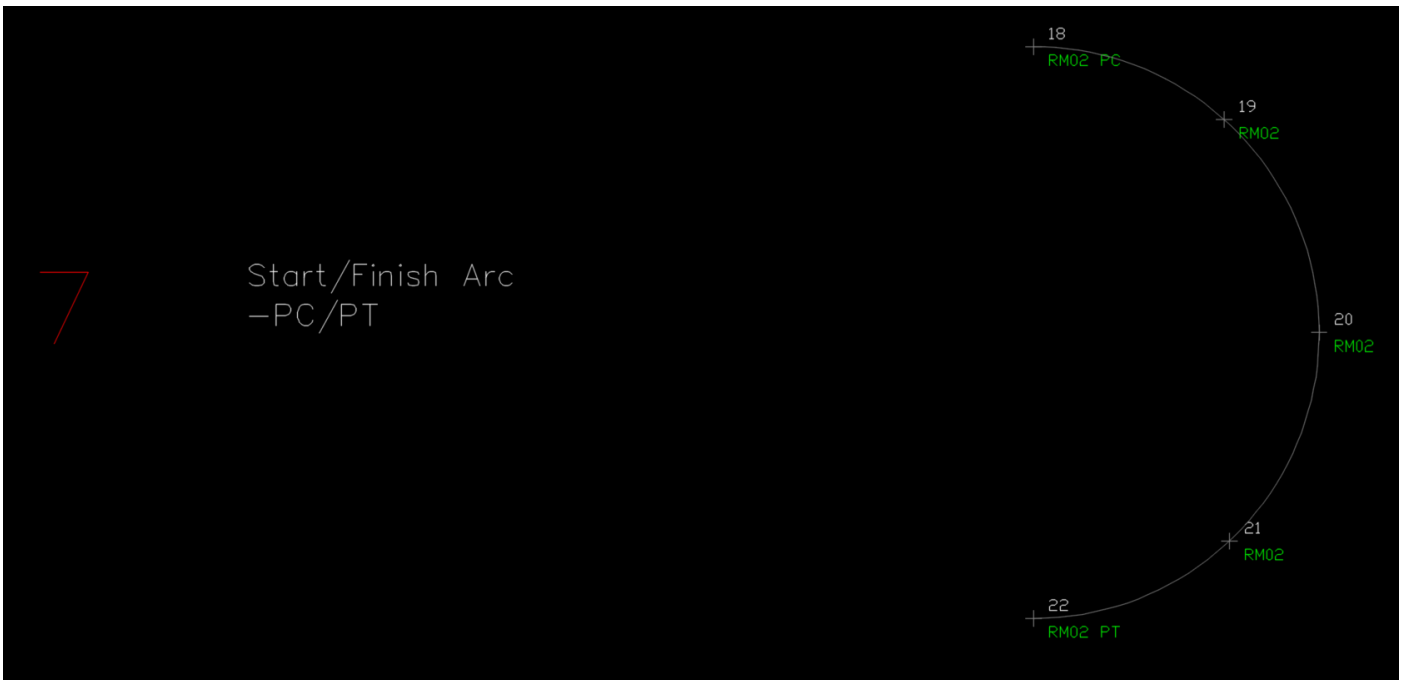
16, 2002, 1992, 0, RM - second point on curve

17, 2000, 1990, 0, RM - end point of curve

*See example next page

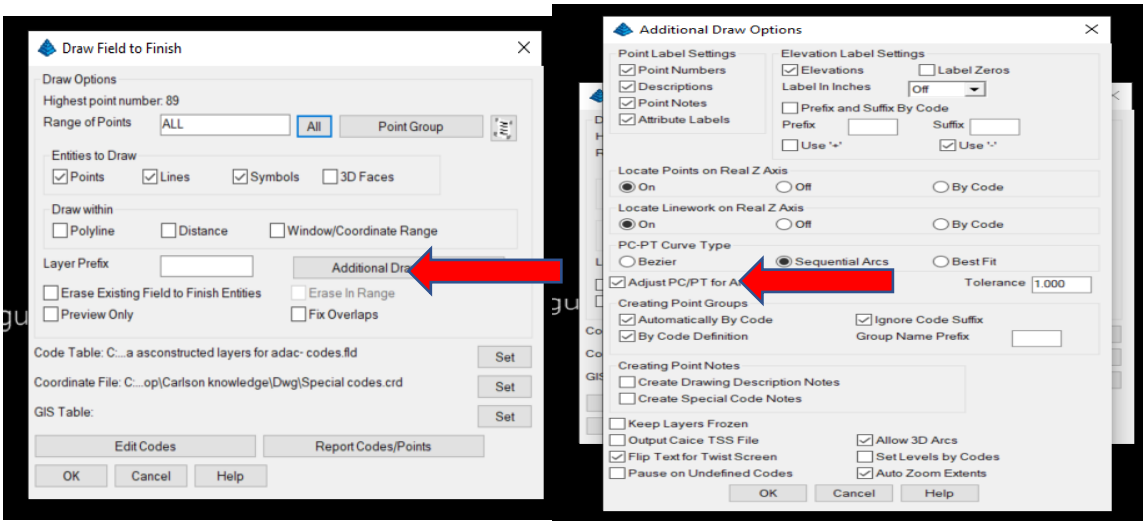
3.9 PT: End Curve (Example 7)

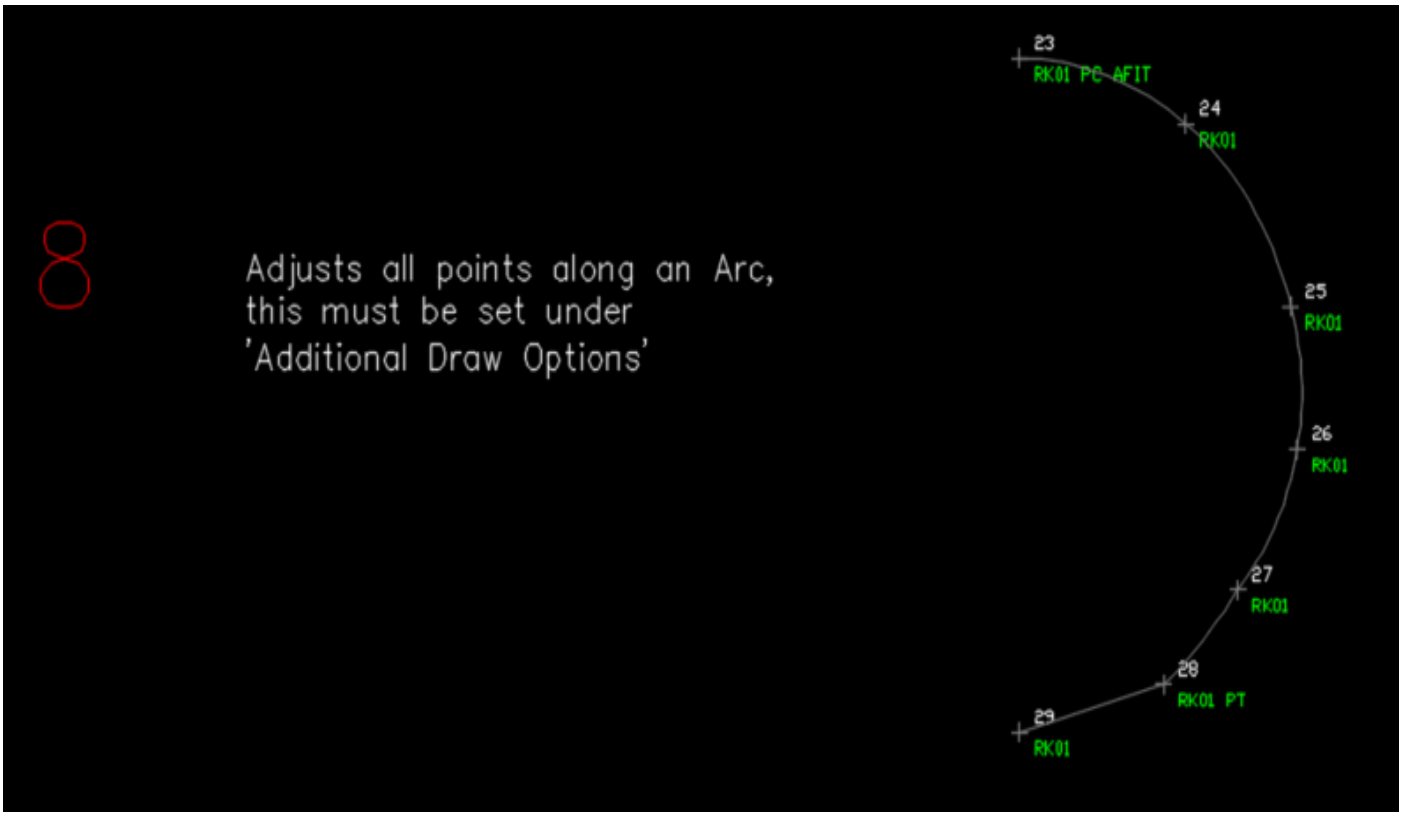
This code can be used with "PC" to define a curve with more than three points or a tangent two-point curve. Starting at the point with the "PC", the program will look for a "PT". If the "PT" is found, all the points between the "PC" and "PT" are used for the curve which is drawn as a smoothed polyline that passes through all points and only curves the polyline between points. If no "PT" is found, then the regular three-point arc is applied as explained above. If no points are found between the "PC" and "PT", then the point prior to the "PC" and the point after the "PT" are used to create tangents for the resulting curve.



3.10 AFIT: Fit Arc (Example 8)

This special code adjusts the PC/PT points for the current arc to make the arc tangential. This special code is a way to individually control this tangential arc adjustment. To adjust all arcs within a specified adjustment tolerance, use the Adjust PC/PT setting under the 'Additional Draw Options' from the first Field-To-Finish dialog.





3.11 BFIT: Best-Fit Line (Example 9)

This code creates a best-fit line using the points for the linework. This feature can be used when you have multiple points on a feature that you know is a straight line such as a footpath and you want a single line to be drawn. Each of the points that you want to include in the best-fit need to have the BFIT code.

3.12 CTOG: Curve Toggle

This special code toggles curve mode on and off. Instead of using PC to start a curve, you can use CTOG. Likewise, instead of using PT to end a curve, you can use CTOG. This only work if the code is a polyline.

3.13 CLO: Close (Example 10)

This code forces the lines drawn between a series of points with the same code to close back to the first point with the same code. For example, shots 35-38 all have the RM03 description with the exception of point 38. Its description is RM03 CLO. This will force the linework drawn for the RM03 code to close back to point 35 which is the first point with the description of RM03.

3.14 GAP: Gap (Example 11)

This special code makes a single segment break in the current linework. For example, if you have a kerb polyline that you want to break to skip over a driveway, then you could add the GAP code at the start of the driveway and continue the kerb as normal on the other side.

3.15 NE: No Elevation (Example 12)

This code represents no elevation. A point with this special code is located at zero elevation.

3.16 YZ: Yes Elevation (Example 13)

This special code locates at the point entities at the coordinate elevation and overrides any other setting for locating the point at zero elevation.

3.17 NOS: Non-Surface

This code indicates that the point should be "non-surface"; that is, that it should be ignored when contouring or creating surfaces. This can also be controlled per-field code by turning on the Non-Surface toggle in the Edit Field Code Definition dialog box.

3.18 ZO: Elevation Only

This code represents elevation only (Z-Only). A point with this special code is used at part of a 3D polyline for elevating the 3D polyline without effecting the horizontal position of the polyline. For example, this code could be used on a grade break point along a cube where only the elevation should change and not the horizontal alignment.

3.19 PHOTO

This code attaches a photo file to the point. The name of the photo file should be right after the PHOTO code. The PHOTO Link setting controls whether the photo is attached using a Carlson-format link or a CAD Hyperlink. Use the Image Inspector command to view photos attached to points by either link method. To use the Hyperlink, you can Ctrl-click or right-click on the point entity.

In addition to the PHOTO code, Field-to-Finish will also automatically create the photo links for SurvCE photos. The program looks for the photo database from SurvCE which should have the same name as the coordinate file with an extension of .phdb. This photo database file should be in a sub-folder of the coordinate file folder and called Pictures_X where X is the name of the coordinate file. For example, if your coordinate file is C:\Projects\Job1\Job1.crd, then the program looks for C:\Projects\Job1\Pictures_Job1\Job1.phdb.

3.20 LABEL:

This code controls the point attribute format using a number after the code. This number uses 0=attribute block, 1=text, 2=both, 3=none. For example, LABEL1 means draw that point using text attributes.

3.21 COLOUR

This code sets the entity colour. The colour is specified after the 'COLOUR' code by either a CAD colour number (1 to 256) or colour letter. The colour letters are R=red, Y=yellow, G=green, C=cyan, B=blue, M=magenta, W=white. For example, COLOUR1 means use CAD colour 1 (red) and COLOURB means use colour blue.

3.22 Offsets: OH, OV, OFL (Example 14a & 14b)

The codes "OH" and "OV" stand for offset horizontal and offset vertical. These offset codes apply to 2D and 3D polylines. A single set of offset codes can be used to offset the polyline a set amount. For example,

49, 2000, 1946, 10, RK04 OH.5 OV-.8

50, 2004, 1946, 11, RK04

51, 2006, 1946, 12, RK04

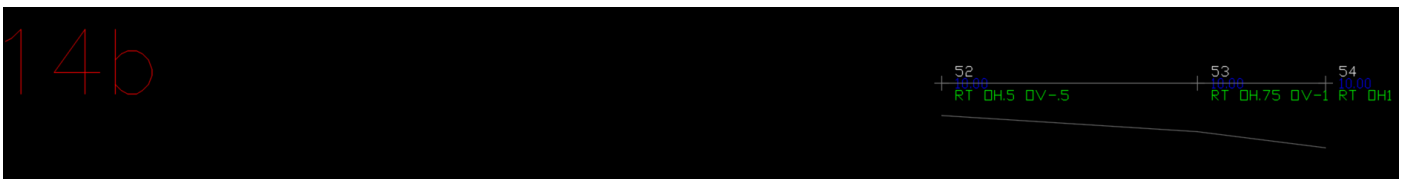
This would create a polyline connecting points 49,50 and 51 and an offset polyline with a 0.5 horizontal and -0.8 vertical offset. The direction of the horizontal offset is determined by the direction of the polyline. A positive horizontal offset goes right from the polyline direction and a negative goes left. The horizontal and vertical offset amounts apply starting at the point with the offset codes until a new offset code or the end of the polyline. Only one horizontal and vertical offset can be applied to 2D polylines. For 3D polylines, multiple offset codes can be used to make a variable offset. For example,

52, 2000, 1943, 10, RT OH.5 OV-.5

53, 2004, 1943, 10, RT OH.75 OV-1

54, 2006, 1943, 10, RT OH1 OV-1.5

This would offset the first point horizontal 2.5 and vertical -0.5, the second point horizontal 5.5 and vertical -0.75 and the third point horizontal 7.5 and vertical -0.75.



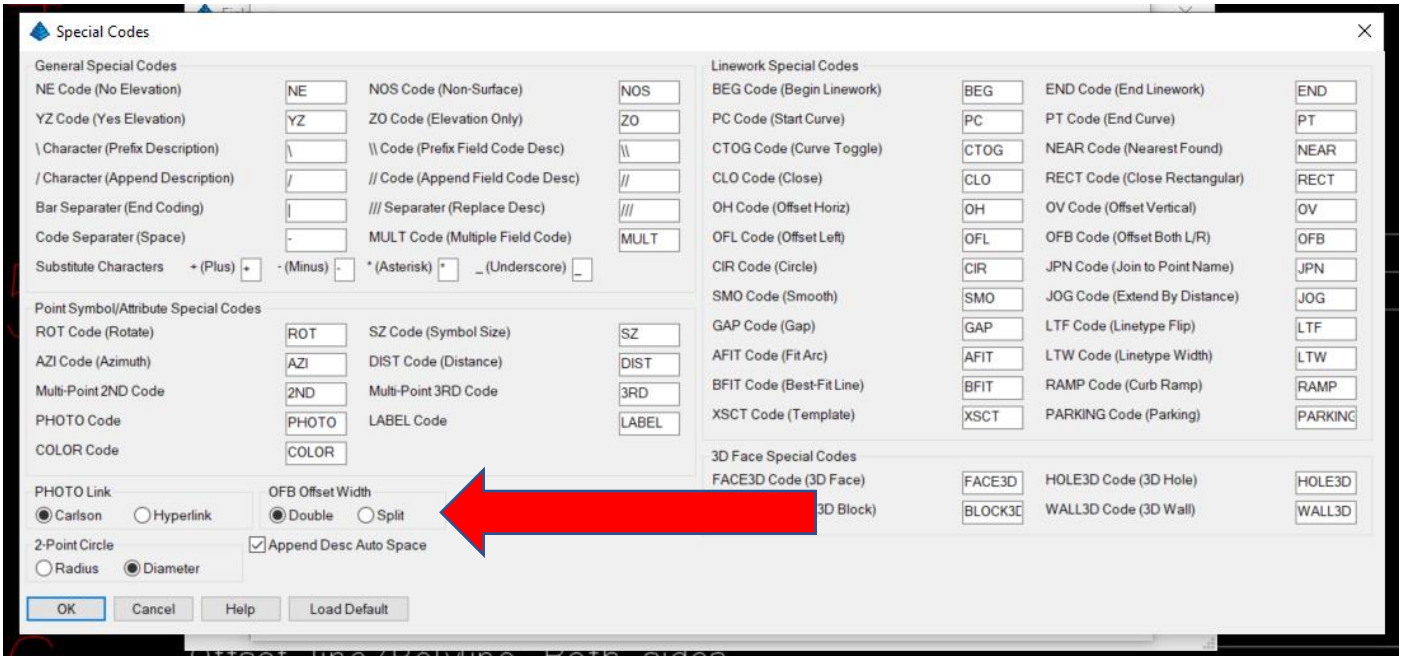
3.22.1 Multiple Offsets (Example 15)

When there are multiple "OH" codes for the same point, the polyline is offset multiple times.

The "OFL" code stands for offset left horizontal. The only difference with the "OH" code is that you don't have to enter the "-" to go left.

3.22.2 Offset both Left & Right (Example 16)

The "OFB" code stands for offset both left and right horizontal. For example, if the points follow the centerline of a road (RCL), the OFB code can be used to create the left and right edges of the road. There is a setting in Special Codes for OFB for whether the offset value is double or half width between the two offset lines. Also, there is a setting in each code definition under the Linetype tab for whether to create a closed polyline from the left and right offset polylines.



3.23 Symbol Size (Example 17)

This code is used to set a different symbol size. There are several ways to use this code. It can take multiple scale factors for different dimensions by putting an ID character after the factor.

SZ: If nothing follows the SZ code, then the next point with the same field code as the current point will be used to determine the size.
SZ#: The value of the new symbol size is specified after the SZ. This value is the actual size in drawing units. For example, SZ2 or SZ3

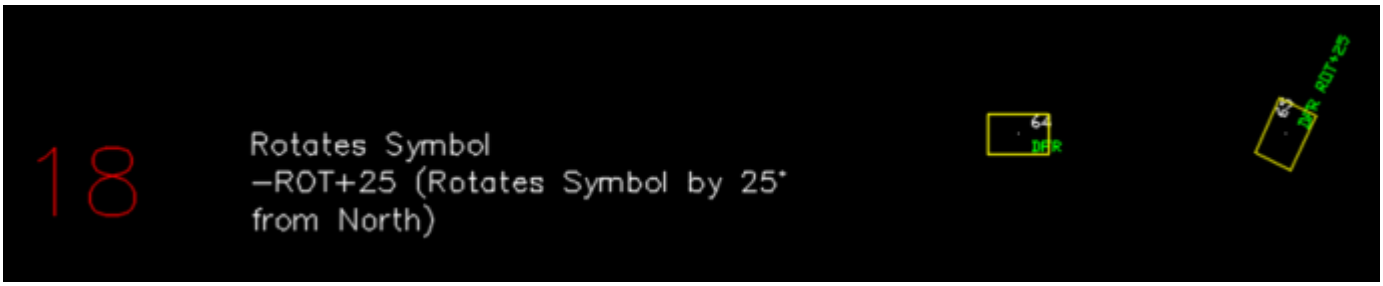
SZ#X: The value after the SZ is used to scale the symbol in the X dimension. For example, SZ2X.
SZ#Y: The value after the SZ is used to scale the symbol in the Y dimension. For example, SZ2Y.
SZ#Z or SZ#V: The value after the SZ is used to scale the symbol in the Z (Vertical) dimension. For example, SZ2Z.
SZ#H: The value after the SZ is used to scale the symbol in the X,Y (Horizontal) dimensions. For example, SZ2H.
SZ#S: The value after the SZ is a symbol size scaler that get multiplied by the drawing horizontal scale to determine the actual drawing units. For example, SZ0.2S.

The X, Y, Z, V and H can be combined. For example, to scale a symbol by 10 horizontally and 25 vertically, use SZ10H25Z. Or to scale a symbol by 2 in the X direction and 4 in the Y direction, use SZ2X4Y.

When multiple SZ codes are used in the same point description, the symbol is drawn multiple times at the different sizes. For example, a point description of "SM SZ5 SZ10" will draw the sewer manhole symbol twice. One symbol will be size 5 and the other size 10.

3.24 ROT: Rotate (Example 18)

This code is used to set the rotation of the point symbol. If a point number follows the ROT code, then angle from the current point to this point number is used for the rotation. For example, "ROT45" would rotate the symbol towards point number 45. If there is no point number after the ROT code, then the rotation point is the next point number with the same code as the current point or a companion code for the current code. ROT can also be used to rotate towards an angle clockwise from north by using `+' or `-' in front of the number. For example, ROT+25 rotates the point symbol to the northeast and ROT-90 rotates the point symbol to the west.



3.25 SMO: Smooth (Example 19)

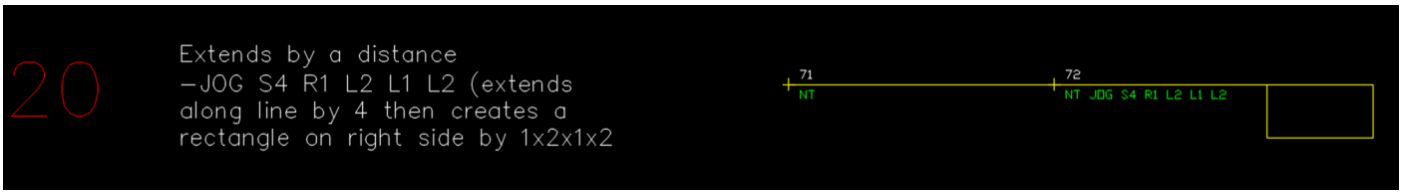
This code is used to smooth the polyline.

3.26 AZI & DIST

The AZI and DIST codes are used together to offset the point. The AZI sets the offset azimuth and DIST sets the distance. The values should directly follow the code. For example, AZI25 DIST4.2 would draw the point offset 4.2 at an azimuth of 25 degrees.

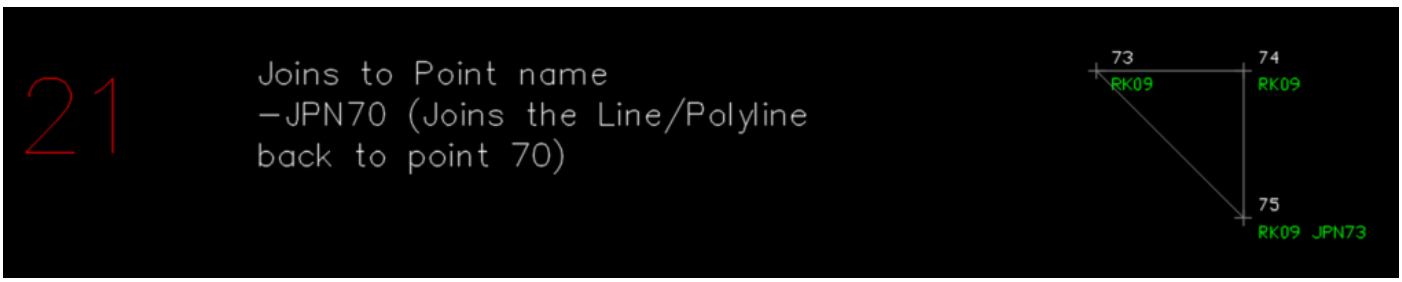
3.27 JOG: Extend by Distance (Example 20)

The "JOG" special code allows for additional points to be inserted into the line work at perpendicular or straight offsets. Only offsets should follow the JOG code. Positive numbers indicate a jog to the right and negative numbers indicate a jog to the left. Alternatively, "R#" and "L#" can be used where # is the distance to either the right or the left. Finally, "S#" can be used to make an offset straight ahead by using a positive # or behind by using a negative #. For example, "NT JOG S4 R1 L2 L1 L2" or equivalently "NT JOG S 5 1 2 1 2" advances 5 units and then draws a closed rectangle on the right-hand side of an existing line. The offsets are always done in the X-Y plane. If the current line is vertical, an offset to the right is along the positive X-axis.



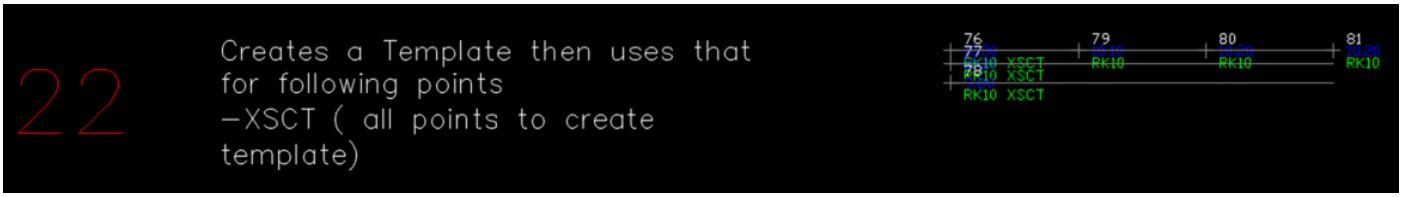
3.28 JPN: Join to Point Name (Example 21)

The "JPN" (Join to Point Name) special code joins to the point named immediately after the code. For example, "JPN73" causes a line to be drawn from the current point to the point "73". JPN is designed to work for adding a segment at the start of linework. So, the point with the JPN code should be at first segment of the linework.



3.29 XSCT: Template (Example 22)

This special code defines a template from a series of points with this special code. This template is then applied like a template defined under Linetype for a code. This XSCT code is a way to define a template in the field instead of having a template with fixed dimensions defined in the code table. For example, you could have a code for RK for back-of-kerb. Then store three points with a description of "RK XSCT" for three points to cross section the kerb: back-of-kerb, top-of-kerb and flow line. After these three "BC XSCT" points, you could have single "BC" points along the kerb and the program will apply the template along these kerb points and draw three parallel lines.

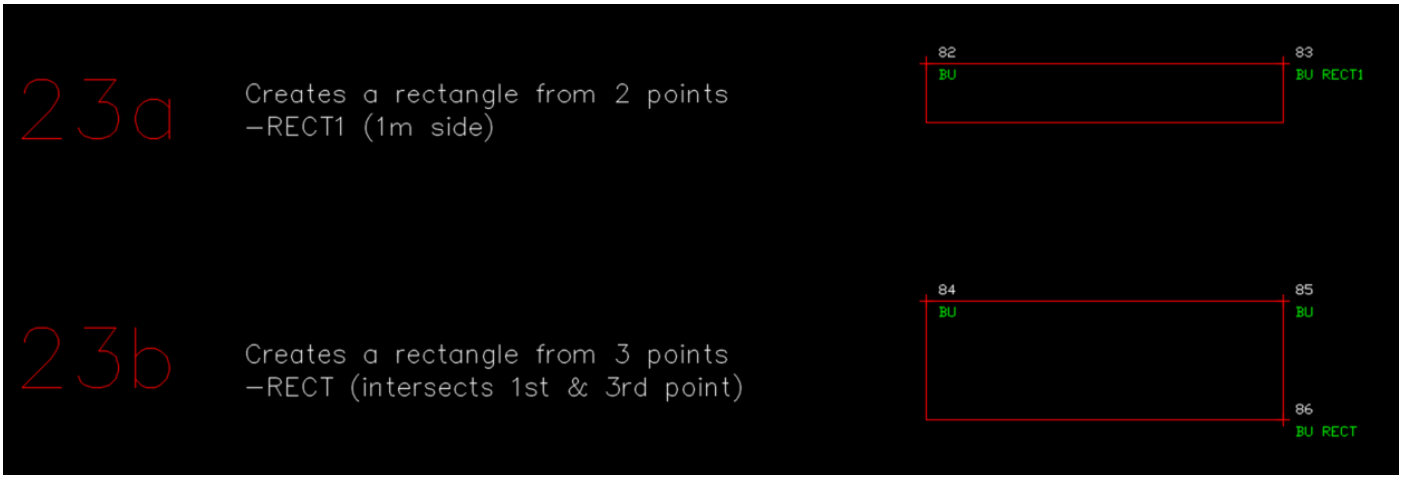


3.30 NEAR: Nearest Found

This special code sets the current polyline to Nearest Found connection order. This applies to codes that have the Connection Order set to Sequential and you want to override this setting to Nearest Found for the current polyline.

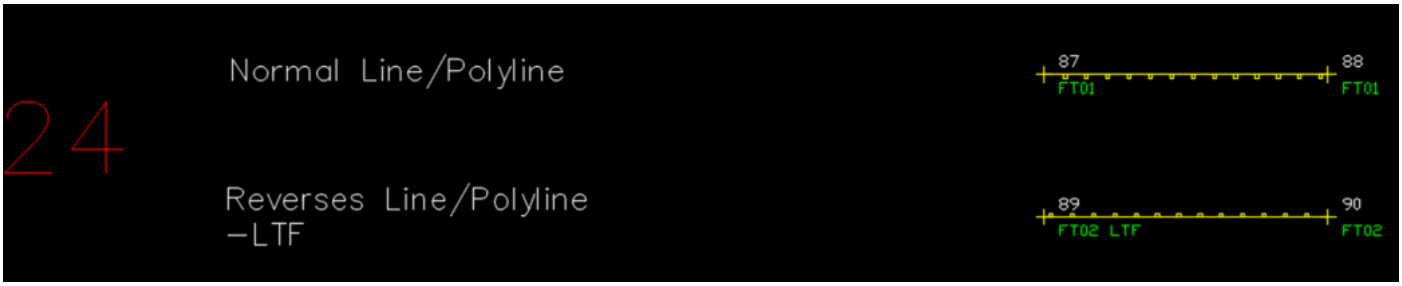
3.31 RECT: Close Rectangular (Example 23a &23b)

The "RECT" special code creates a rectangle as a 2D or 3D polyline using one of two different methods. If a number follows "RECT" (e.g., "RECT10"), a rectangle will be drawn 10 units to the right of the last two points ending on the point with the "RECT" code. Use a negative offset to place the rectangle on the left side (e.g., "RECT-2.5"). For example, if locating the left side of a 10m rectangular concrete pad using the code conc for concrete, the description of the two left points would be (conc) for the first point and (conc rect10) for the second. If no number follows "RECT", then the polyline will be closed by shooting right angles from the first point of the polyline and the current point and creating a new point where those two lines cross. This method requires three points be established on the rectangle. In this method, the "RECT" code can be on any of the polyline points.
*See next page for example



3.32 LTF: Lintype Flip (Example 24)

This special code switches the side for the linetype. This option applies to non-symmetrical linetypes like the treeline or guard rail for when you want the linetype to face the other way.

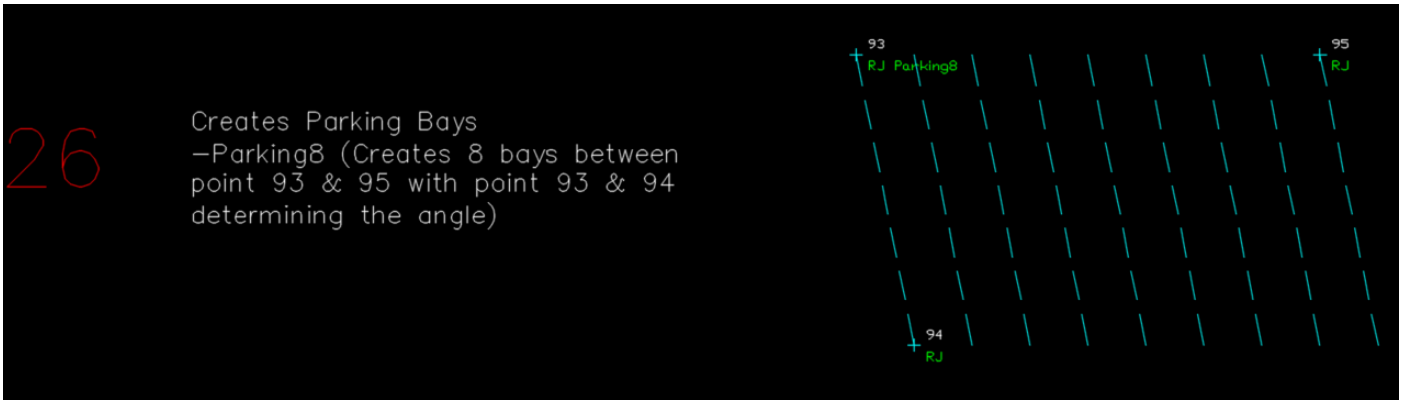


3.33 LTW: Line Width (Example 25)

This special code sets the line width. The width value is entered after the LTW code. This width is applied to lines and 2D polylines.

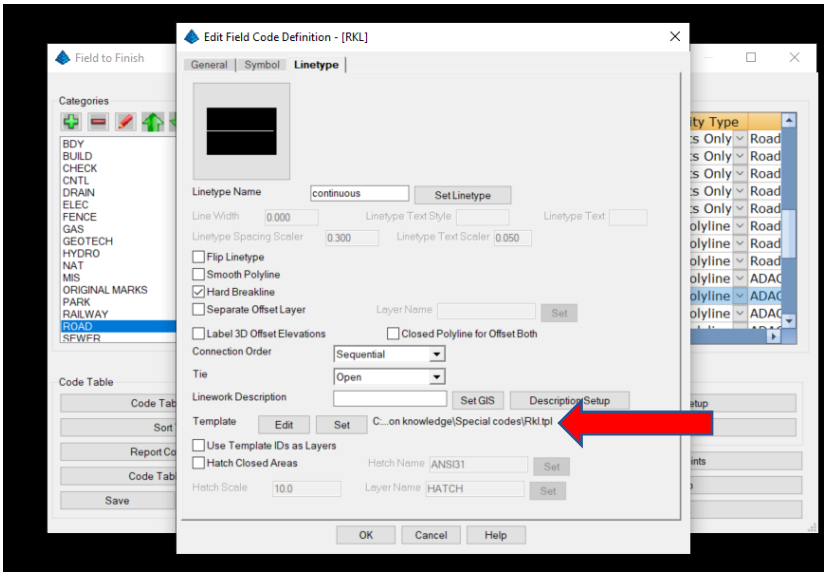
3.34 PARKING: Parking (Example 26)

This special code draws parking stall lines using three points. Points one and two are used to draw the first parking line and define the length and angle of the lines. The third point defines that position of the last parking line. A number needs to follow the PARKING code for the number of parking lines to draw. For example, PARKING8.

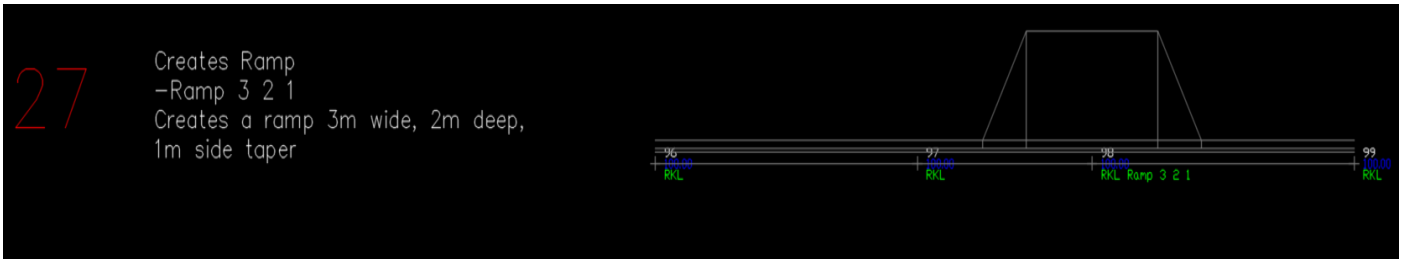


3.35 RAMP: Kerb Ramp (example27)

This special code adjusts the 3D polylines to create a kerb ramp. The routine looks for parallel 3D polylines for the bottom of kerb, top of kerb and back of kerb. These 3D polylines can be created in Field-To-Finish by using a Template under the Code > Linetype definition (if using the Template method than that needs to be set)



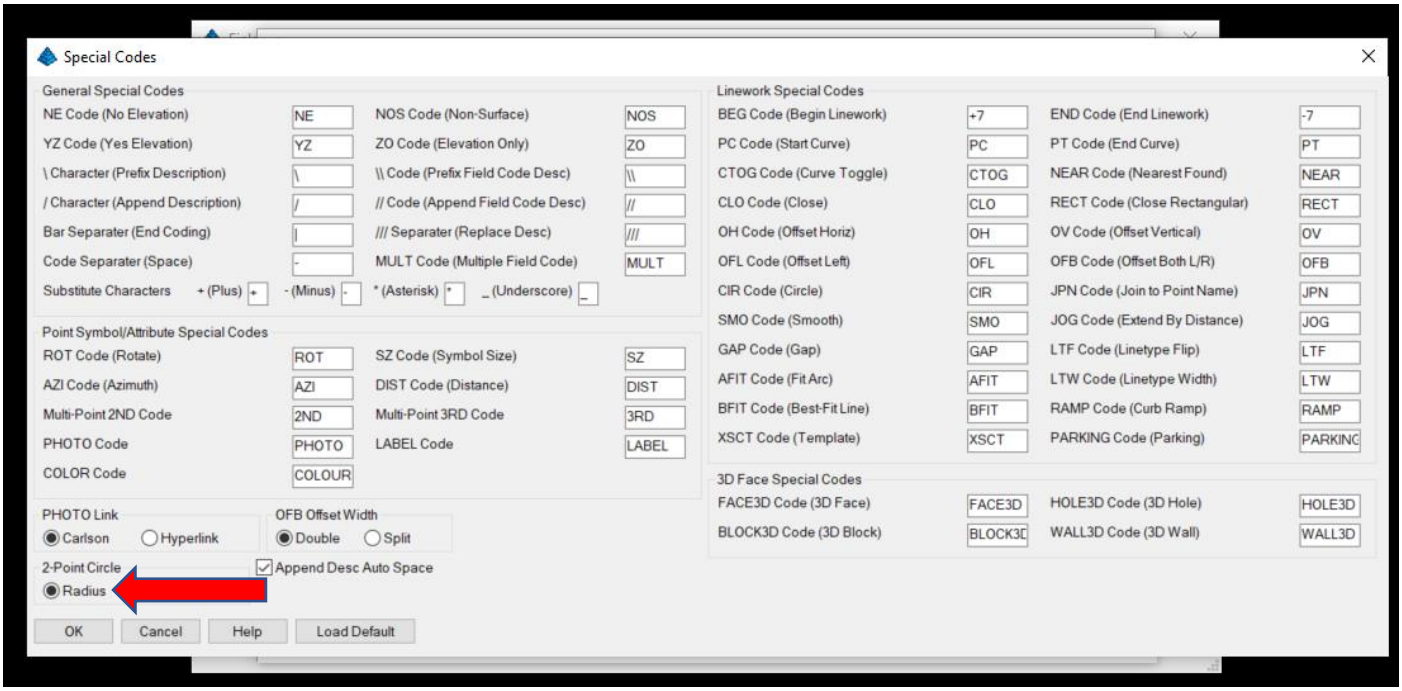
The lines can be also created using the offset special codes, or by having three separate 3D polylines. The ramp is centered at the point with the RAMP code. The dimensions of the ramp follow the code in order of width, depth and taper. For example, a description of "RKL RAMP 3 2 1" uses "RKL" for a bottom of kerb code then "RAMP" for the special code and 3 for the width, 2 for the depth and 1 for the side taper



3.36 CIR: Circle: (Example 28 & 29)

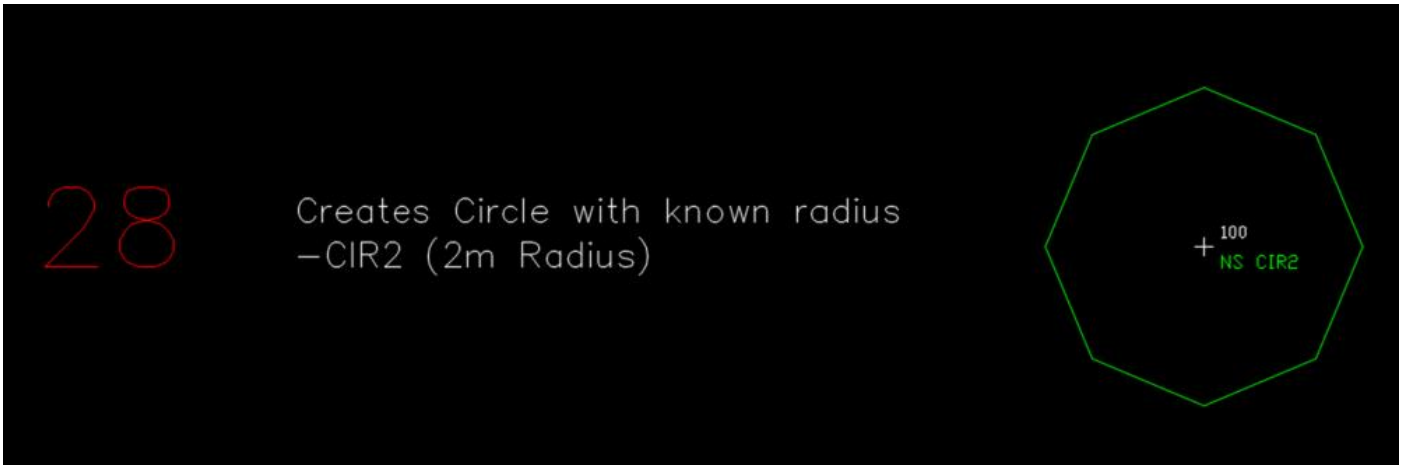
The "CIR" special code causes the point to create a circle in one of three different ways. The first way uses just the current point as the center with the CIR special code followed immediately by the radius. For example, "CIR2" will create a circle centered on this point with radius 2 and at the elevation of the current point. The second method uses two points, the first point specifying the center and the elevation, and the second point specifying the radius. Only the first point has the "CIR" special code and the second point is the next point with a matching field code. Another variation of method 2 is using 2 points that are on the perimeter and define the diameter. For the 2 point method, whether the points define the radius or diameter is defined on the Code Table Settings > Special Codes dialog. The third method uses 3 or more points that specify the perimeter of the circle in 2D with the first point specifying the elevation. For this method, the "CIR" special code is only on the first point and the rest of the points are the next points with matching field codes.

The "CIR" code can be used with all of the linetypes including "points only". The circles are always parallel to the X-Y plane unless the code linetype is set to "3D Polyline". Then the circle is drawn as a 3D polyline. Any active linework for the code is ended before processing the "CIR" special code.
The next 2 methods can only be achieved if '**Radius**' has been selected in the 2-Point Circle section of Special Codes

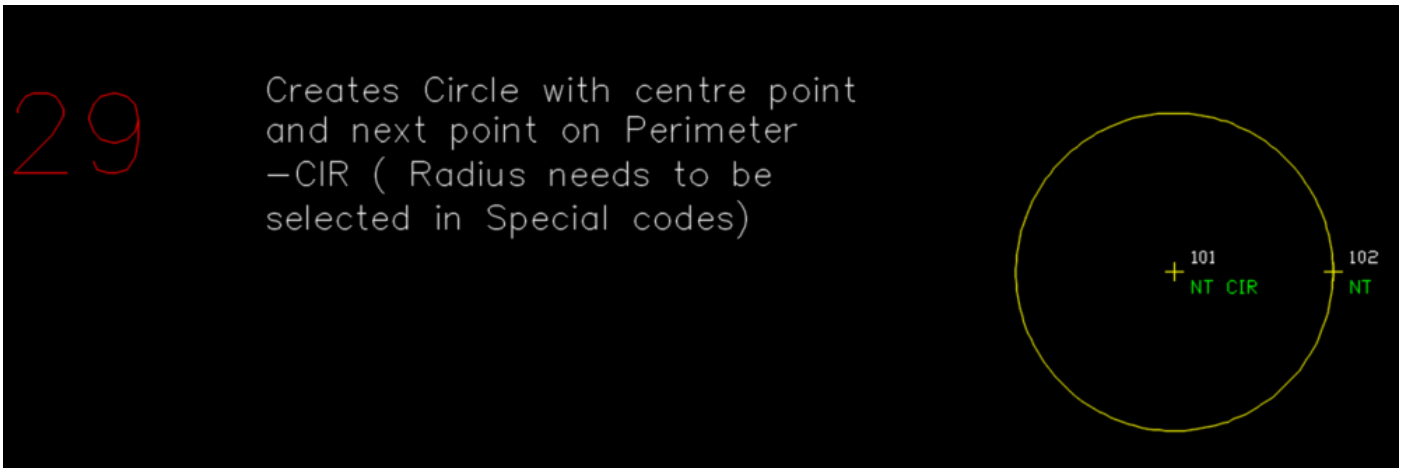


Method 1 (Single point at center with radius value(2m)) **(Example 28)**

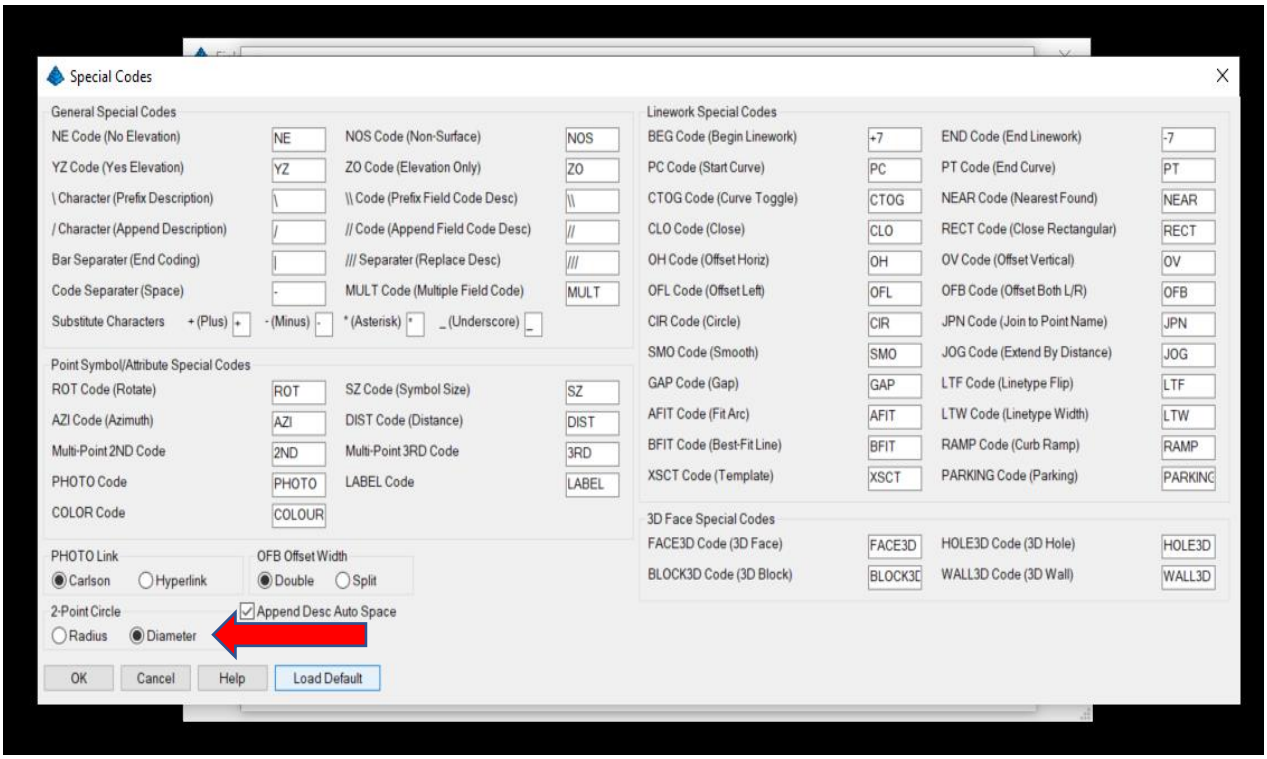
100 NS CIR2



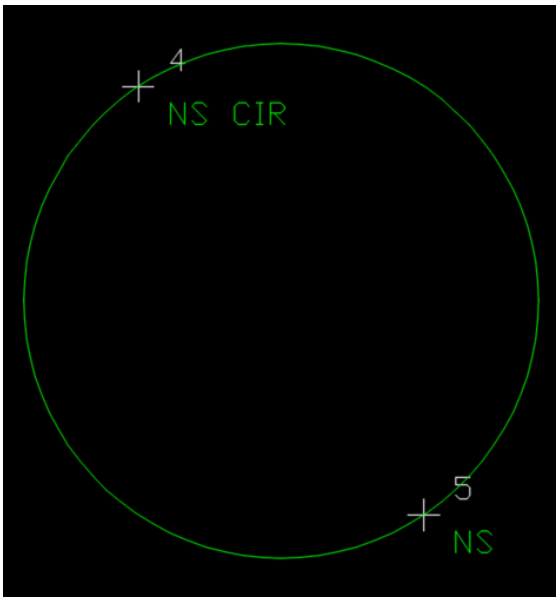
Method 2 (Point at center plus point at perimeter) **(Example 29)**
101 NT CIR
102 NT



The following method can only be achieved if '**Diameter**' has been selected in the 2-Point Circle section of Special Codes.



Method 3 (2 Points on perimeter that define the diameter)
4 NS CIR
5 NS



Method 4 (Points on perimeter) can be achieved if either radius or diameter is selected. If doing more than one circle then strings need to be used to separate the different circles.

8 BC01 CIR
9 BC01
10 BC01