



Walt Disney World Swan and Dolphin Resort  
Orlando, Florida

## Intersection Design with Autodesk® Civil 3D®

Daniel Philbrick - Autodesk  
and James Wedding (Assistant); Chakri Gavini (Assistant)

**CV41-1L** Get an in-depth look into the practices and techniques for creating intersections using Autodesk Civil 3D. The Corridor modeling and grading tools will be explored in this interactive lab.

### **About the Speaker:**

Daniel is a software development manager for the Civil Engineering products developed in Autodesk's Infrastructure Solutions division. In this role, he has managed the design and implementation of the Autodesk Civil 3D product since its inception. Dan joined Autodesk in 1998 and has managed the development of several releases of Autodesk Land Desktop, Autodesk Civil Design, and Autodesk Survey. He has an MS degree in Mechanical Engineering from the University of New Hampshire. Prior to joining Autodesk, Dan worked in computational fluid dynamic software development at Northern Research and Engineering Corporation and at Pratt & Whitney.

**[daniel.philbrick@autodesk.com](mailto:daniel.philbrick@autodesk.com)**



## Course overview

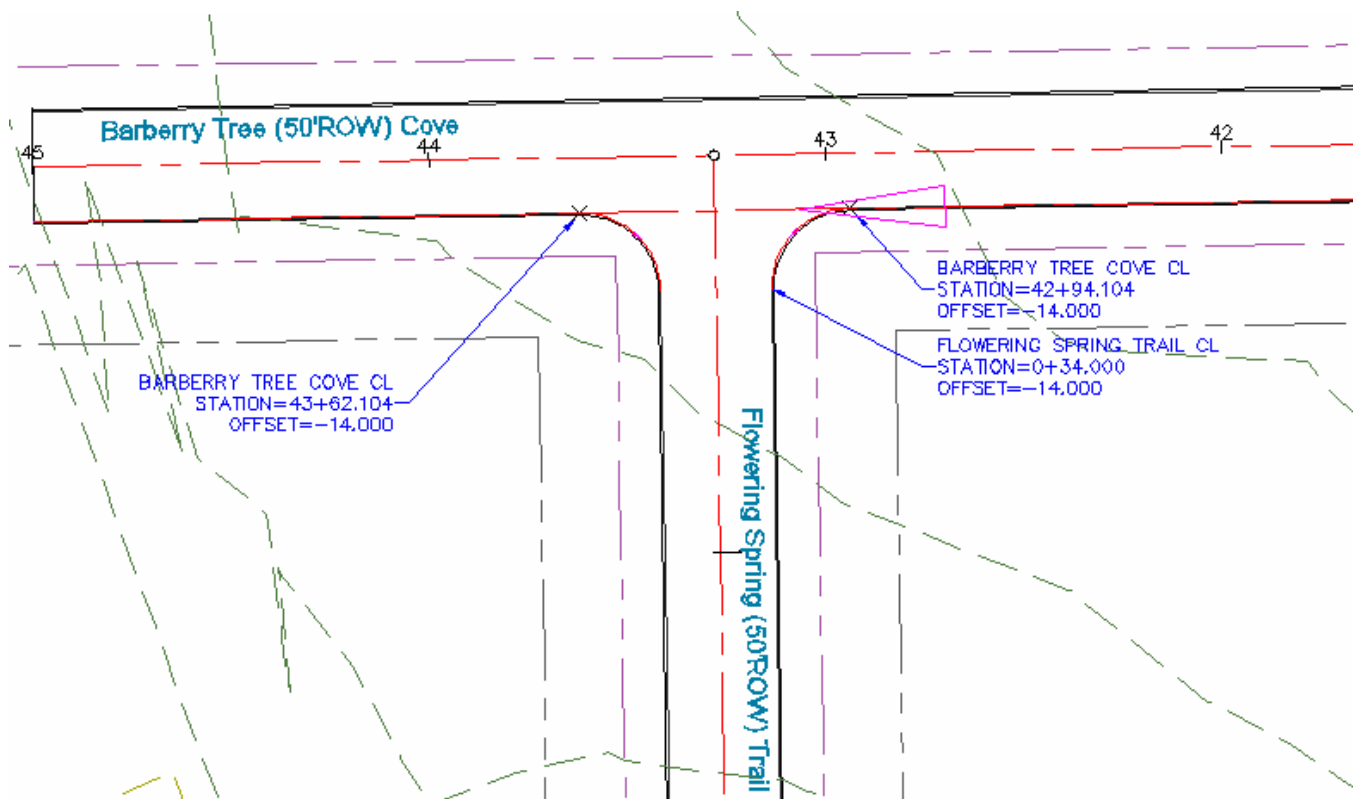
In this lab we are going to explore various methods for creating intersections in Autodesk Civil 3D. We will be leveraging features from alignments, profiles, corridors, sample lines, and sections as we proceed through this hands-on lab.

The techniques for modeling the intersections depend on the type of intersection which is being designed, and constraints within the design problem. Intersections for secondary roads will have different considerations than intersections for high speed travel ways. Also, the drainage requires of the intersection also need to be considered in the design.

The goal of this Lab is to familiarize the attendee with the functionality in Autodesk Civil 3D through a hands-on intersection design session. We are going to design a classic "T" intersection using 3 different techniques, each of which have their specific role for in a particular design.

## Dataset – Starting point of the design

The data set consists of a "T" intersection. The alignment running horizontally is called **Barberry Tree Cove CL** and the alignment intersecting vertically is called **Flowering Spring Trail CL**. The profiles for each of these alignments have also been designed.



The Curb return horizontal design and profiles have also been established as a baseline. The Curb Return on the left is called **Flowering Spring West ROUNDING** and the Curb Return on the Right is called **Flowering Spring East ROUNDING**. The starting and ending elevations of the Curb returns match the elevation of the lane.

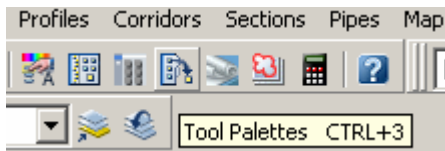
We will begin by just familiarizing ourselves with the dataset.

1. File → Open and Navigate to the file **CV41-1L\Intersection Baseline Design.dwg**
2. Take a few minutes to locate the 4 alignments and their associated profiles. The Profile Views have already been created in the Drawing.

## Subassemblies – Establishing design parameters

Subassemblies are the building blocks of the Corridor design. Before we begin the intersection design, we are going to create a tool palette and add the specific assemblies we are going to use. We are also going to set the specific parameters on each of the subassemblies. This process can be done in a blank new drawing.

3. File → New
4. If the Tool palettes are not already visible, select the tool palette icon from the toolbar



5. Next create a new palette and call the Palette ***Intersection Design***.

We are now going to create the specific tools for this design.

The subassemblies which will make up the typical design assembly include:

BasicLane

BasicCurbandGutter

BasicShoulder

BasicSideSlopeCutDitch

BasicLaneTransition

6. We will now set the specific parameters for the subassemblies and add them to the new tool palette. This is done by selecting the previously listed

subassemblies, drag them into the drawing, and select return to place them in the drawing. The AutoCAD Properties Manager can then be used to set the specific parameters on the Subassemblies. Drag each of those Subassemblies into your drawing window.

7. Select the **BasicLane** and change the Advanced parameters to the following:

The screenshot shows the Properties Manager for the BasicLane subassembly. The 'ADVANCED' tab is selected, and the 'Parameters' section is expanded. The parameters are as follows:

Parameter	Value
Version	R2006
Side	Right
Width	14.0000'
Depth	0.6700'
Slope	-3.571%

8. Select the **BasicShoulder** and set the parameters to the following:

The screenshot shows the Properties Manager for the BasicShoulder subassembly. The 'ADVANCED' tab is selected, and the 'Parameters' section is expanded. The parameters are as follows:

Parameter	Value
Version	R2006
Side	Right
Width	3.0000'
Depth	0.6700'
Slope	2.000%

9. Select the **BasicLaneTransition** and set the Advanced parameters to the following:

The screenshot shows the Properties Manager for the BasicLaneTransition subassembly. The 'ADVANCED' tab is selected, and the 'Parameters' section is expanded. The parameters are as follows:

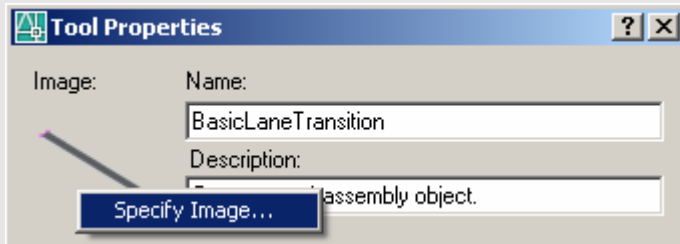
Parameter	Value
Version	R2006
Side	Right
Default Width	14.0000'
Depth	0.6700'
Default Slope	-3.571%
Transition	Change offset and elevation

Make sure to set the Transition option to "Change offset and elevation".

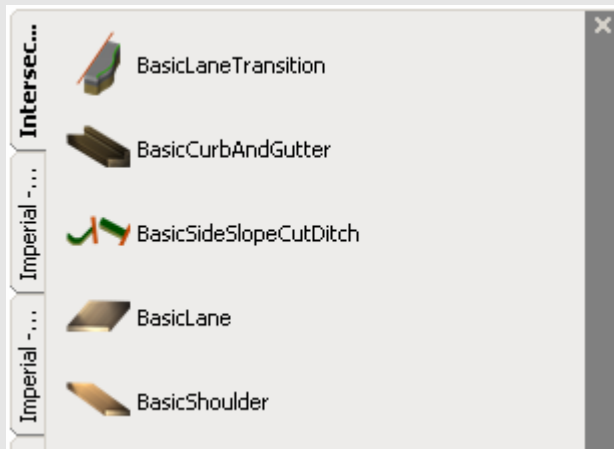
10. Leave the defaults for the **BasicCurbandGutter** and the **BasicSideSlopeCutDitch** subassemblies.

Now that we have created modified versions of the Subassemblies, we are going to drag them onto the new tool palette.

11. Select the Subassembly from the drawing so that you have it highlighted, with a grip. Now reselect it and drag the Subassembly from the drawing, onto the tool palette. It will create an instance of the tool on your new tool palette.
12. The image for the tool is set by **selecting tool -> right clicking -> Properties**. On the properties dialog, **select the image, Right click, specify image**. You can then browse to the image location and set the appropriate images.



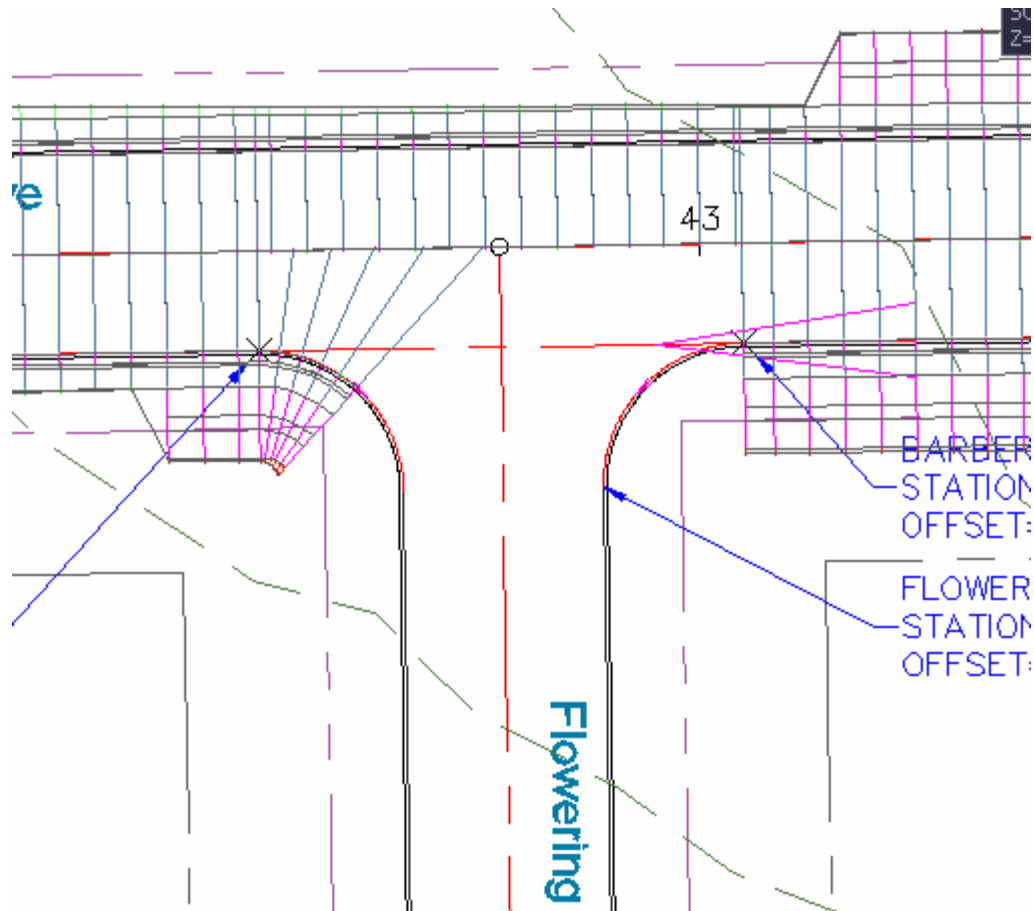
Your tool palette should now contain 5 subassemblies and look something like:



## Designing the intersection

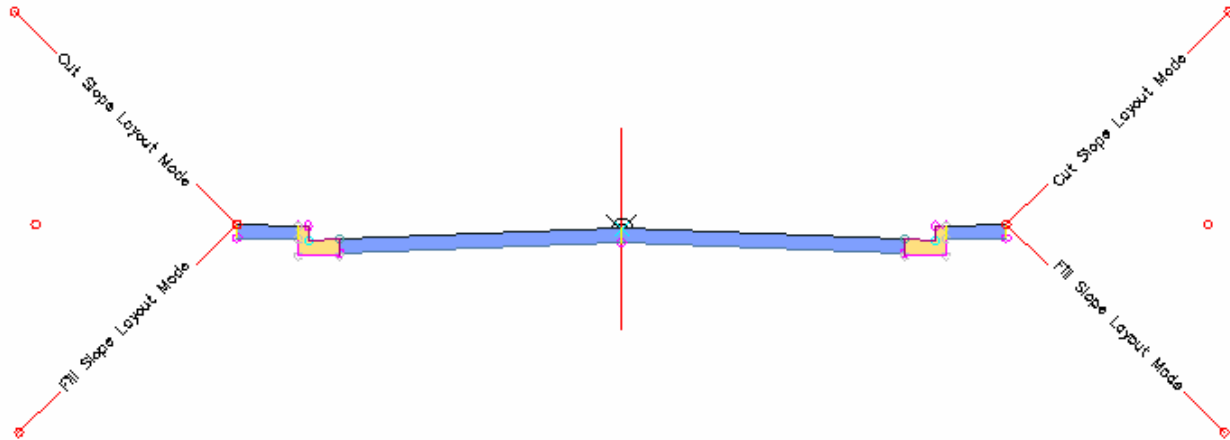
In the first version of the intersection, we are going to build a single corridor model, with multiple baselines. The first baseline is going to be Alignment running horizontally across the design. This will have 3 regions – the region before the intersection, the region within the intersection and the region after the intersection.

Each of the curb returns will also be a baseline. We will design this so that the lane transitions from the Curb return baseline, to the centerline of the through road. The design will have four **Baseline** Alignments. The 2 intersecting alignments and the 2 curb return alignments make up the four baselines. The following figure illustrates a partial solution to show the various regions.



## Creating the Assemblies

In the next step, we will create the typical assembly for use in the design. The typical Assembly for the through portion of the main road will look like the following:



### Typical Assembly for the Intersecting Alignments

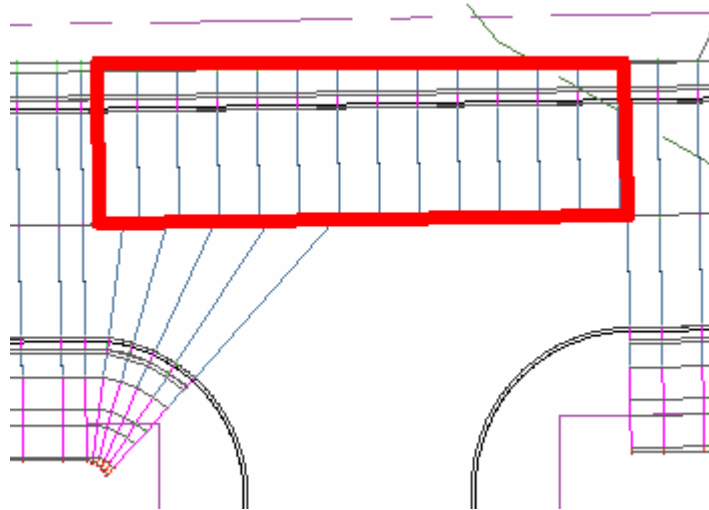
13. First, create an Assembly by selecting **Corridor -> Create Assembly** from the menu. Create the typical assembly section by attaching the 1) BasicLane 2) BasicCurbandGutter 3) BasicSidewalk 4) BasicSideSlopeCutDitch subassemblies. You should use the subassemblies which we added to the new tool palette. You should end up with a typical Assembly which looks like the proceeding illustration.

Naming the Assemblies is important so that you know where they should be applied in the design. You should select a naming convention so that you can understand. Naming them according to where they are applied in the various corridor regions is a good strategy.

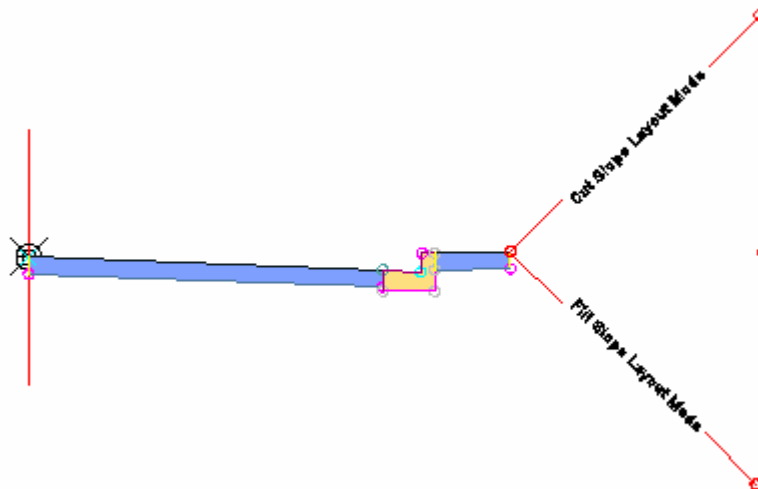
14. Select the Assembly, Right Click and change the Assembly name to **Intersection Typical Assembly**.

Next we are going to create the assembly for the portion of the through road which is part of the intersection. This will effectively be the same as the typical assembly, with the left had side of the assembly removed. This will be used in the area highlighted in Red below.



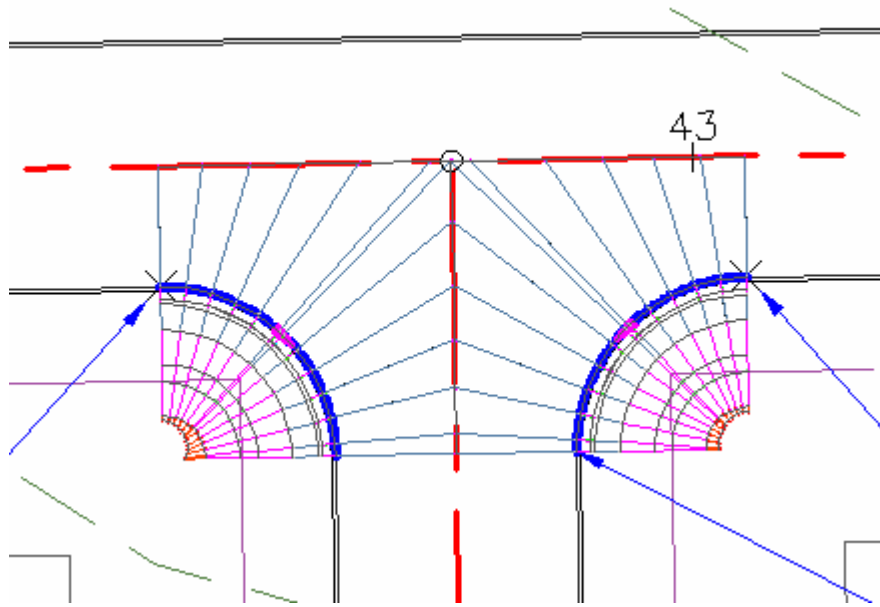


15. This can be created by copying the previous Assembly and deleted all the subassemblies on the Left hand side. To do this, select the Assembly which was just created, **Right Click and select Basic Modify Tools -> Copy**. It will then prompt you for a placement location. Let's also change the name of this Subassembly to **Barberry Tree Cove CL Through Lane**. After deleting the left hand side subassemblies, you should have an assembly which looks like the following:



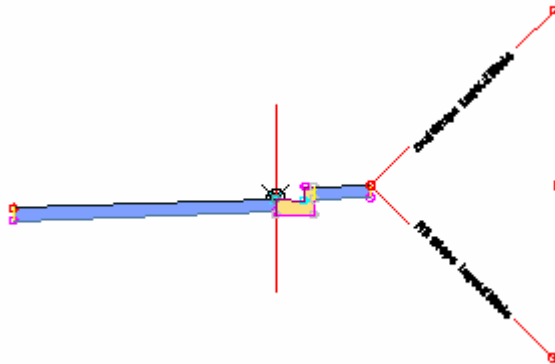
Barberry Tree Cove CL Through Lane

The next step is to build the Assemblies which will be used on the Curb Returns. These subassemblies will attach to the Curb Return alignments and the Lane portion will transition back to the Centerline Alignment and Centerline Profile of the intersecting roads. The following figure illustrates the strategy for these Assemblies.



Because the alignments are running in opposite directions we will create 2 assemblies for these regions.

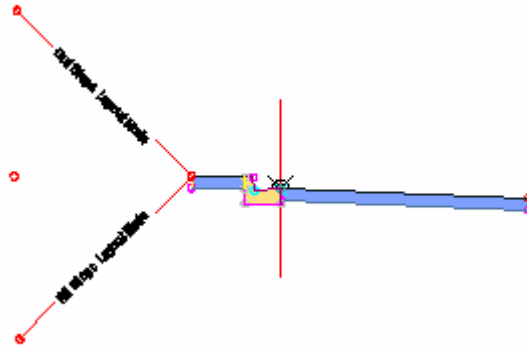
- For the Baseline called **Flowering Spring West ROUNDING** the Lane is going to be on the left side of the Assembly and the Curb and Gutter, Shoulder and Side Slope Grading will be on the right. Create a new assembly and add the **BasicLaneTransition** to the left side, and the **BasicCurbandGutter**, **BasicShoulder**, and **BasicSideSlopeCutDitch** to the right side. Edit the Subassemblies name and change it to **Flower Spring West Rounding**. Your Assembly for this baseline should look like the following:



Flowering String West Rounding Subassembly

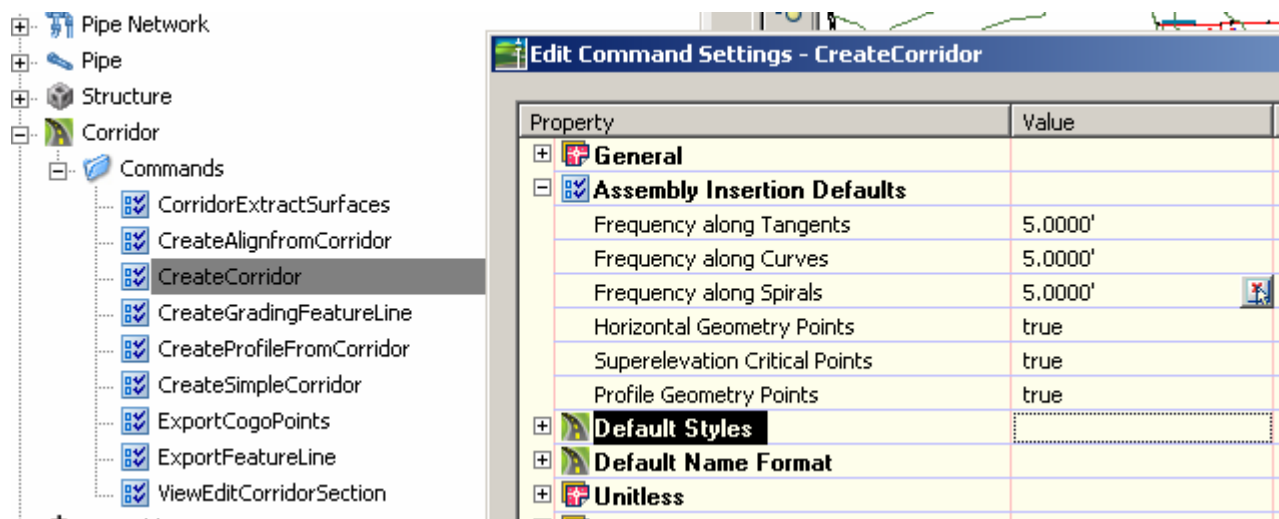
- Lastly, the subassembly for the fourth baseline, **Flowering Spring East ROUNDING** needs to be created. For this baseline, the lane transition needs to be on the right hand side of the

Assembly. This is basically a mirror of the previous subassembly. Create the fourth assembly using the **BasicLaneTransition** on the right side, and the **BasicCurbandGutter**, **BasicShoulder**, and **BasicSideSlopeCutDitch** applied to the left side. Rename this assembly to **Flowering Spring East Rounding**.



Flowering String East Rounding Subassembly

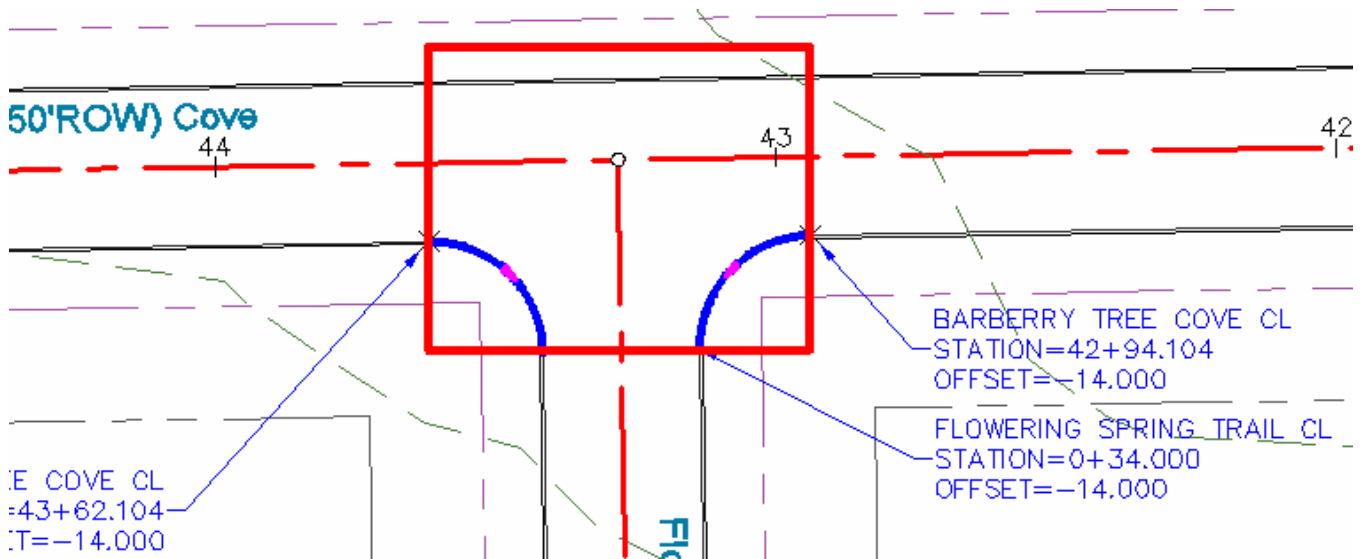
18. Lastly, let's set a few defaults. The first is the sampling increment for the Corridor and the second is the layer for the corridor. First go to the settings manager, Corridor Node, and select Edit Feature settings. Edit the Sampling Increment and set it to **5'**.



19. Now select the drawing node in the Setting Manager and Right Click and select the **Edit Drawing Settings**. Select the Object Layers Tab and verify that the layer is set to **Corridor**. When we begin creating a Corridor, it will now go to the Corridor layer.

## Creating the Intersection – Method 1

The first step will involve creating a Corridor and applying the Assembly called **Intersection Typical Assembly**, outside the intersection region. These regions include the areas outside the Red box shown below.



We will now start building the Corridor. The Corridor will consist of 4 baseline alignments, each with multiple regions. The first step will be to apply the Insertion Typical Assembly on the Barberrry Tree Cove CL Baseline. The Regions will be as follows:

Baseline 1 – Barberrry Tree Cove CL

Region 1: Starting Station = 42+00, Ending Station = 42+94.104

Region 2: Starting Station = 43+62.104, Ending Station = 44+00

Baseline 2 – Flowering Spring Trail CL

Region 1: Starting Station = 0+34.000, Ending Station = End of Alignment

20. Select the command **Corridor -> Create Simple Corridor...** from the Menu. Name the Corridor **Barberrry Tree Cove Intersection**. Select the **Barberrry Tree Cove CL** Alignment and the **Barberrry Tree Cove (FG)** Profile. Select the **Intersection Typical Assembly** for the Assembly. Set the Logical Name for the surface to **ng** on the Logical Name dialog.

Logical Name	Object Name	Subassembly
Surfaces	<Click here to set all>	
TargetDTM	ng	BasicSideSlopeCutDit...
TargetDTM	ng	BasicSideSlopeCutDit...
Alignments		
Profiles		

This results in a Corridor which runs the length of the main alignment. Select the Corridor and set the Starting and ending stations to:

Region 1: Starting Station = 42+00, Ending Station = 42+94.104

Next create a new region on the Corridor Properties -> Parameters Tab.

Name	Alignment
Baseline (1)	Barberry Tree Cove CL
Region (1)	

Insert Region...

Remove Region

Copy to Clipboard

Refresh

Set the starting and edit stations for Region 2 and also set the Logical names for the Surface Side Slope Grading.

Region 2: Starting Station = 43+62.104, Ending Station = 44+00

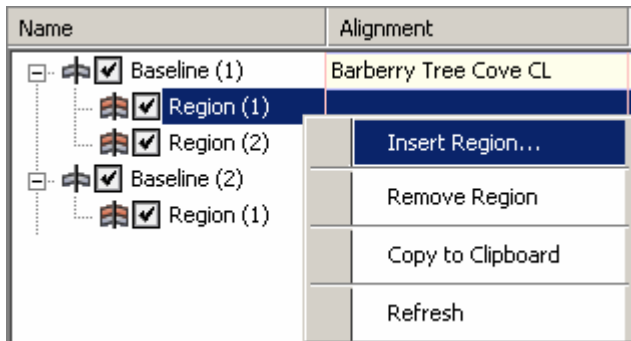
21. Now we are going to add the Second Baseline which is on alignment Flowering Spring Trail CL. This Baseline also uses **Intersection Typical Assembly** and the regions is defined as follows:

Baseline 2 – Flowering Spring Trail CL

Region 1: Starting Station = 0+34.000, Ending Station = End of Alignment

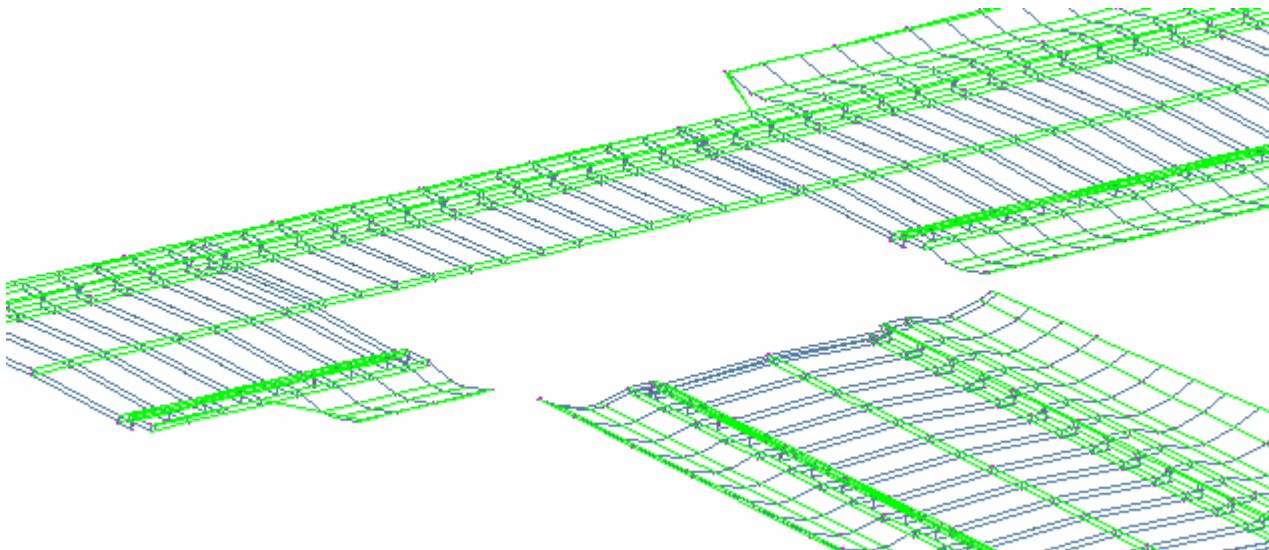
Now we are going to fill in the section of the intersection which runs along the through alignment. Since it is a requirement to have the stations increasing as regions are added to a baseline, the Assembly called **Barberry Tree Cove CL Through Lane** needs to be inserted in between the 2 regions on Baseline 1.

22. Select the Corridor and edit the properties. Select Region 1, Right Click and Select Insert Region:



Select the Assembly called **Barberry Tree Cove CL Through Lane** and set the stations to the beginning of the intersection region, and the end of the intersection region. Before leaving that dialog, set the Logical name for the surface to **ng**.

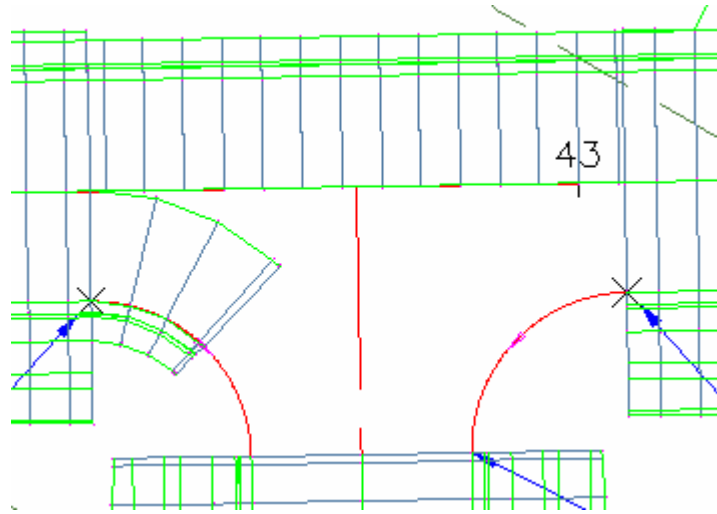
The Corridor should now look like the following.



The next step is to fill in the Curb Returns. We will be treating the Curb Returns as additional baselines. Each curb return baseline will have 2 regions – the first region will transition to the main alignment, the second baseline will transition to the intersecting alignment.

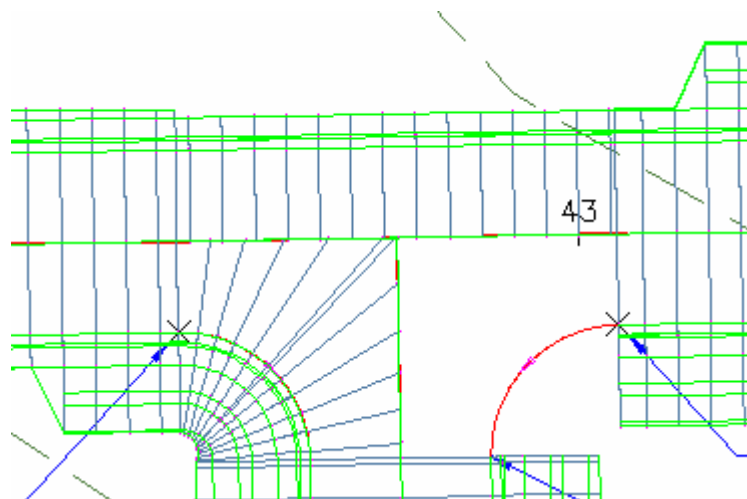
23. Select the Corridor and edit the Properties. Select the Add Baseline button and Select the Alignment called **Flowering Spring West ROUNDING** and the Profile called **Flowering Spring West**

**ROUNDING EOP FG.** Select the End of the region to be the intersection of the two main alignments. You should have something that looks like the following:



Now we need to set the logical names for the lanes so that the lanes transition to the main alignment.

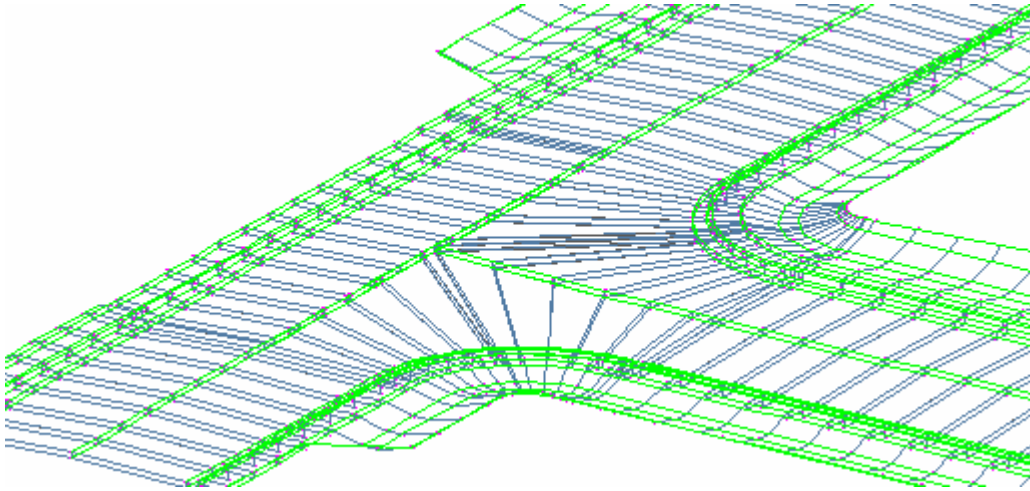
24. Select the Corridor, Edit the properties and Select the Logical name button for the Region which was just created on the Curb Return. Also set the modeling Frequency to 3' along the curves. At the Same time, add the second region. Make sure to set the logical name to the intersecting alignment for the Lane Transition. The WEST curve return should look like the following:



Lastly, the same steps need to be repeated for the EAST ROUNDING Curb return.

25. Repeat the same steps for the Second Curb Return. Make sure to select the Assembly called **Flowering Spring East Rounding**.

The final intersection should look like the following in a 3D view:



You will find a drawing called **Intersection Example 1 – complete.dwg**.

## Creating the Intersection – Method 2

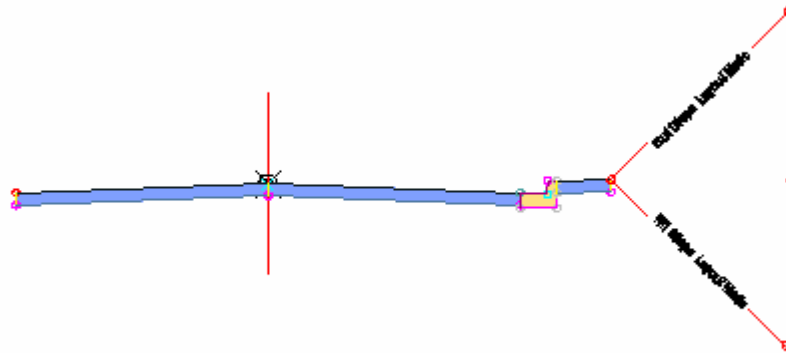
In the previous example, the lanes in the curb return regions transitioned to the centerline of the main alignment. The type of design may not be desirable, particularly for secondary roads interchanging with high speed roads.

In the next example, we will complete an intersection where the curb returns transition to the edge of pavement. This will keep the crown of the road constant through the main road. We will need to create a short alignment along the main alignment EOP.

We are also going to build off the design which we completed in the previous example. If you were not able to complete the previous example, you can start from the drawing called **Intersection Example 1 – complete.dwg**.

26. We will need a new Assembly which will be used in the intersection region of the main alignment. The Right side of the Assembly is exactly the same as the previous example, the left had side need to have the BasicLane added. To create this new Assembly, Zoom to the Assembly called **Barberry Tree Cove CL Through Lane**. Select the Assembly, Right Click and select Basic Modify Tools -> Copy. Drop the copy below the other Assemblies we have already created. Rename the Assembly to **Barberry Tree Cove CL Through Lane – Example 2**. Lastly, add the BasicLane Subassembly to the Left side of the Assembly. Your Assembly should look like the following:

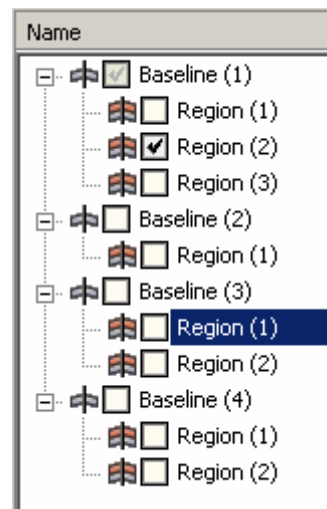




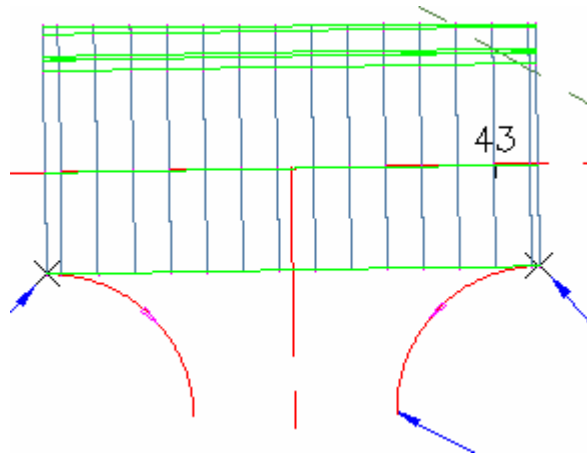
### Barberry Tree Cove CL Through Lane – Example 2

We are now going to build on the Corridor in the previous example. The key change is to use the newly created Assembly in the intersection, Export a short Alignment Segment which represents the EOP, then transition the curb returns to the EOP Alignments.

27. Select the Corridor in the previous example and disable the modeling of all regions except for the region through the intersection. Your Corridor Properties should look like the following:



Next, change the Subassembly for the second region on Baseline 1 to **Barberry Tree Cove CL Through Lane – Example 2**. You will also need to reset the logical names for the Side Slope Grading. Your Corridor should look the following:

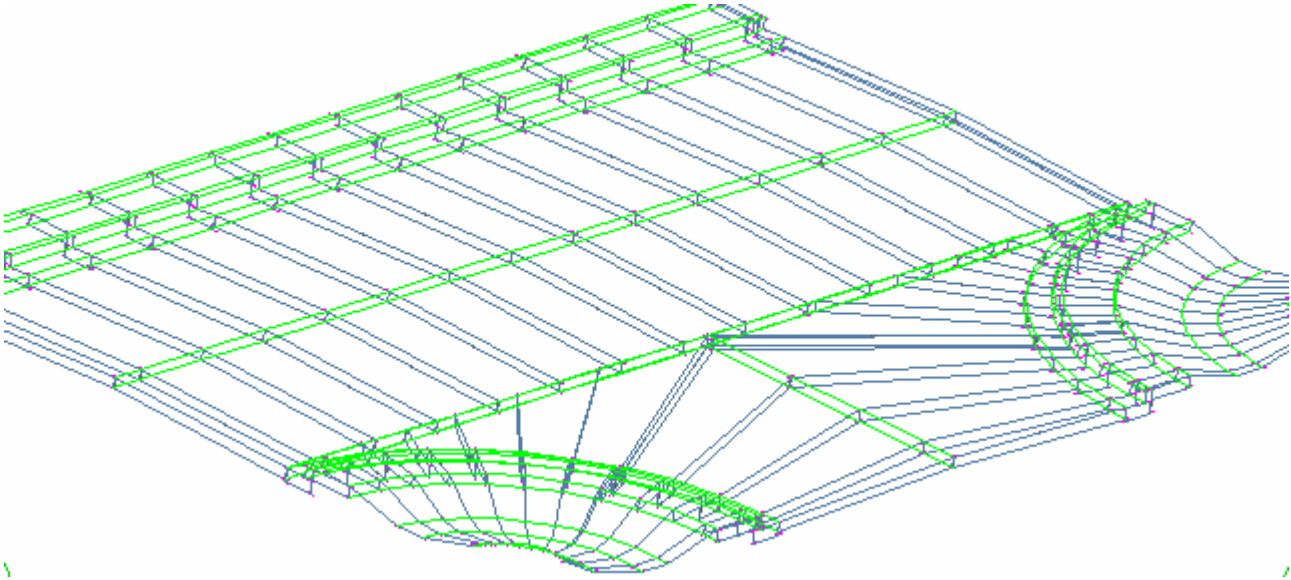


We need to now create an alignment which will be used for the transition alignment for the Curb Returns. Select (from the Menu) **Corridor -> Export -> Corridor Feature Line as Alignment**. Select the Left hand side EOP, the ETW feature line, and name the **Barberry Tree Cove – EOP**. On the profile dialog, name the profile **Barberry Tree Cove - EOP FG**. This results in a short alignment segment, with a profile, representing the EOP of the left hand travel way.

Now that we have an Alignment representing the Edge of Pavement of the lane, we can transition the Curb returns to this EOP alignment and profile. These regions have already been established in our previous design, however we need to change the location of the regions and the logical names for the Transitional alignments.

28. Select the Corridor and Enable **Baseline (3)** and **Baseline (4)**. This will result in LaneTransitions back to the CL of the main alignment. Edit the Corridor properties and Select appropriated starting point, ending point, and logical name for each of the regions on the Curb returns.

If you have set everything appropriately, you should end up with a Corridor which looks like the following:



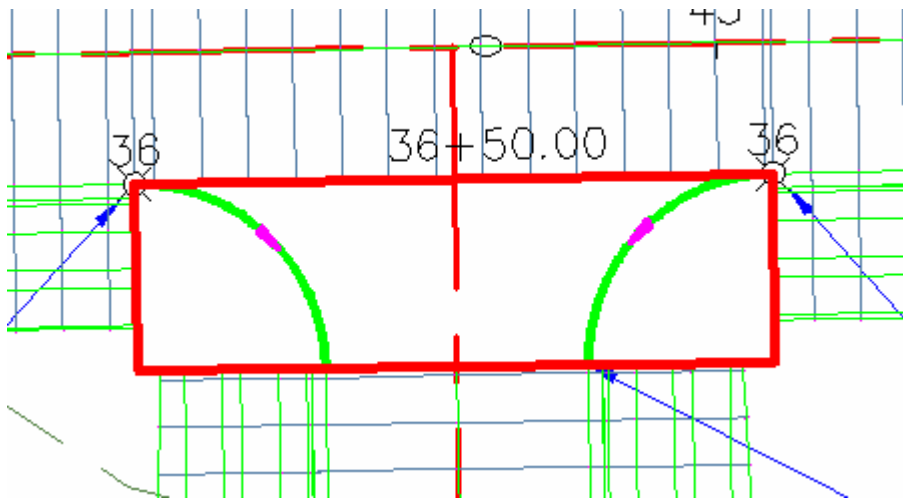
Lastly, enabling the regions outside the intersection will result in the conclusion of this portion of the lab. This design is effective since you have a consistent crown on the main alignment and can control the drainage of the secondary road via the Profile of the Curb return.

There is a file called **Intersection Example 2 – Complete.dwg** which you can refer to as a complete design.

## Creating the Intersection – Method 3

In the next section, we are going to design the intersection such that the Lanes remain perpendicular to both alignments. The lane cross slope of the secondary road will be attached to the alignment and profile of the curb return alignments. The side slope grading will then be perpendicular to the curb return alignments.

The figure below illustrates the strategy for this corridor design.

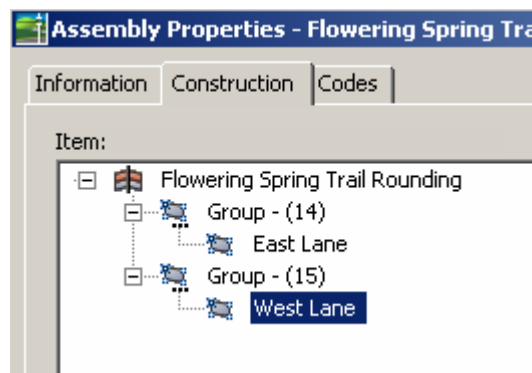


The first step is to create a subassembly to work in the curb return region of the Alignment. For this region, we will create a single Assembly which is referencing the Alignment **Flowering Spring Trail CL**. The lane will transition from the Centerline to the Curb Return Alignments. Next, we will create Offset baselines. The offsets will reference the Curb Return Alignments.

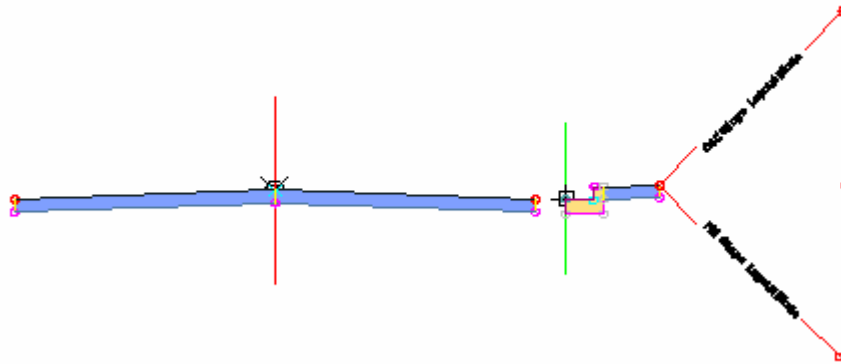
29. Open the drawing called **Intersection Example 3 – Start.dwg**. Create a New Asembly and name the assembly **Flowering Spring Trail Rounding**. After creating the Assembly, attach the BasicLaneTransition to both side of the Assembly.

Since we are going to be applying a single Assembly in this region, naming the Subassemblies will be important in the logical name assignment process. The Left hand side of the Assembly is going to be applied to the East Side and the Right hand side of the Assembly to the West side.

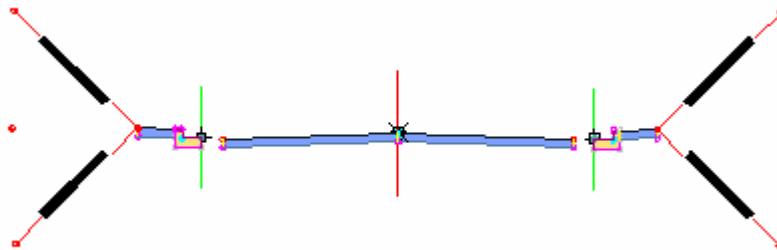
Edit the Assembly and change the name of the Subassembly on the Left side to **East Lane** and change the lane on the Right side to **West Lane**. Your assembly properties should look like the following:



Next create an assembly offset. Select the Assembly, Right Click and Select **Add Offset**. Select a location to the right of the lane on the right side. Now add the BasicCurbandGutter, BasicShoulder, and BasicSideSlopeCutDitch to this offset assembly. Edit the Assembly properties and change the name of this offset to **West Offset**. You should have an assembly which looks like the following:

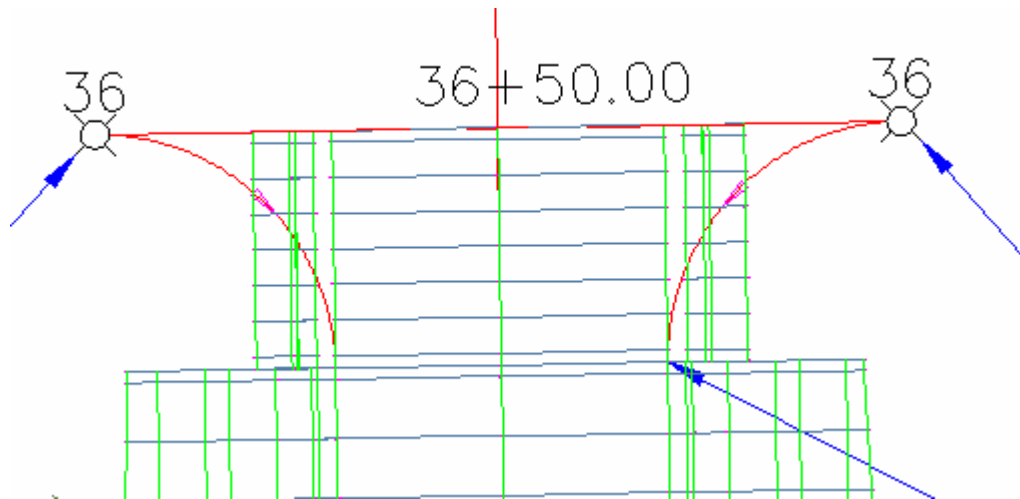


Repeat the same set of steps to create the Offset on the left side of the Assembly. Make sure to rename this offset (in the assembly properties) to **East Offset**. The final Assembly should look like the following:



The final step is to add this assembly to the Corridor in the intersection region.

30. Start by selecting the corridor and disable the modeling of Baseline 1. Add a new baseline and Select the Alignment **Flowering Spring Trail CL** and the FG profile. Next, insert a region and select the assembly called **Flowering Spring Trail Rounding**. Select the starting point and ending point for this Assembly – the starting point is the EOP of the main alignment, the ending point is the ending station for the Curb Return alignments. Applying this to the corridor will result in the following:



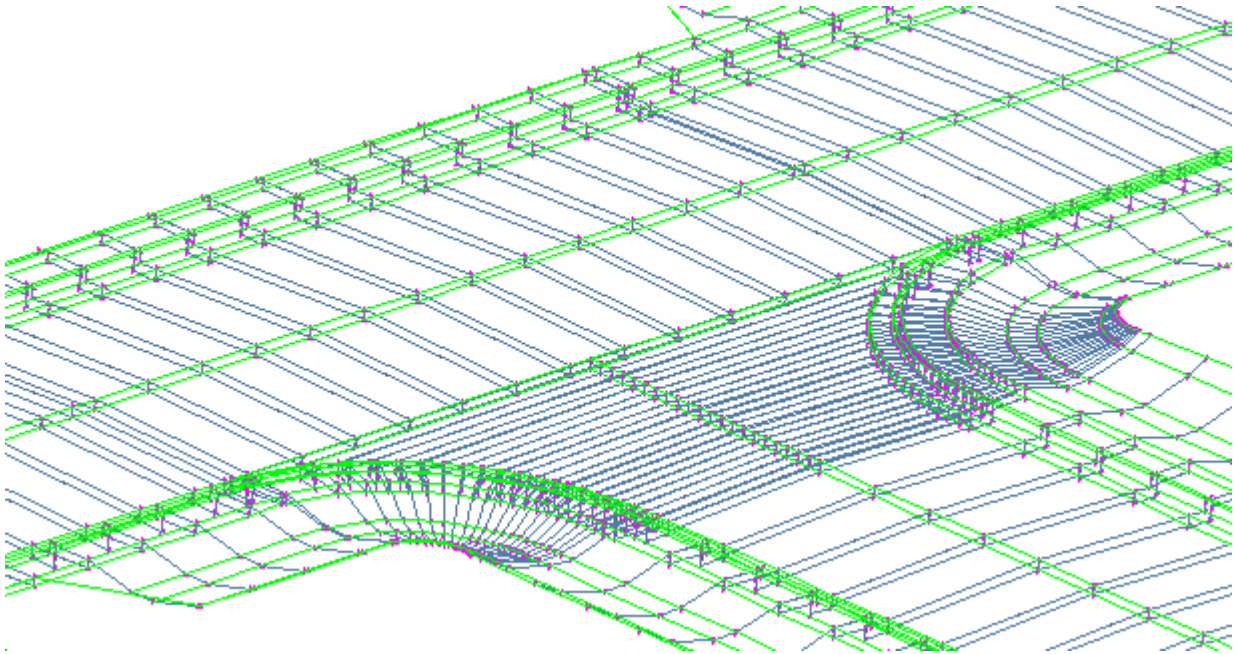
Notice that the lanes are not transition to the curb returns, and the offsets defined in the Assembly are applied.

31. Edit the corridor properties and set the logical names for the lane transitions. The logical name assignments are as follows:

Logical Name Mapping			
Corridor name: Barberry Tree Cove Intersection			
Assembly name: Flowering Spring Trail Rounding		Start Station: 0+14.07'	End Station: 0+34.10'
Logical Name	Object Name	Subassembly	Assembly Group
[-] Surfaces	<Click here to set all>		
TargetDTM	ng	BasicSideSlopeCutDit...	Group - (16)
TargetDTM	ng	BasicSideSlopeCutDit...	Group - (17)
[-] Alignments			
TargetHA	Flowering Spring West ROUNDING	West Lane	Group - (14)
TargetHA	Flowering Spring East ROUNDING	East Lane	Group - (15)
[-] Profiles			
TargetVA	Flowering Spring West Round E...	West Lane	Group - (14)
TargetVA	Flowering Spring East Round E...	East Lane	Group - (15)

32. The offset baseline now needs to be set to the Curb return alignments. Set these to the respective alignments.

You will now have an intersection which looks like the following (the increment for the tangent section of the Assembly was changed to 1'. Also, you may find that you have a Gap in the corridor if you snapped to the starting station which exactly corresponds to the beginning of the curb return. The reason for this is due to the intersection of the BasicLaneTransition is not found with the Curb Return Alignments. Changing the starting station slightly in the direction of increasing station resolves the issue. The final design should look like the following in a 3D view.

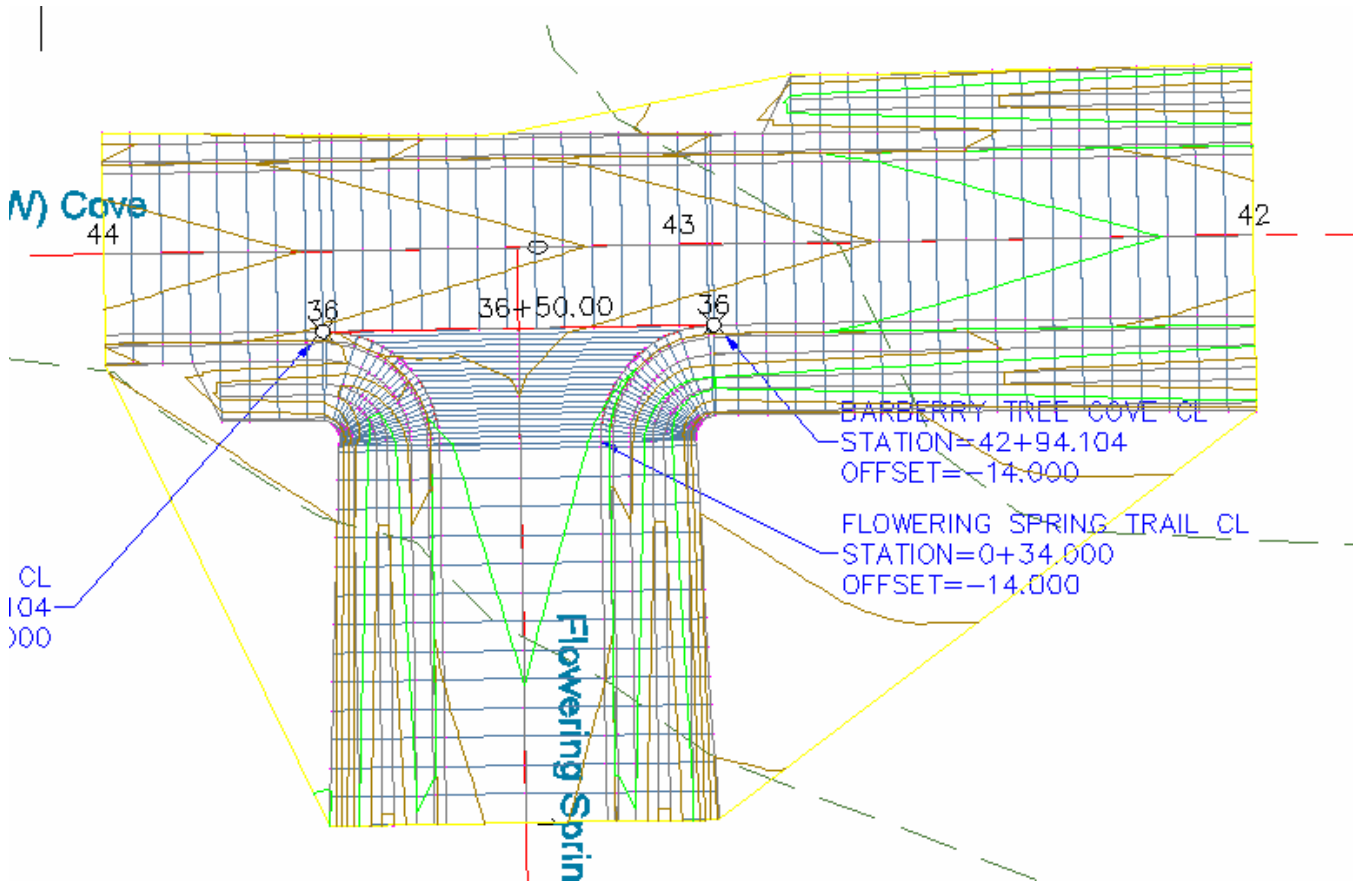


If you have had difficulty in generating the Corridor, you will find a file called **Intersection Example 3 - Complete.dwg** which has a Corridor of the intersection portion of the design.

## Building a Top Surface

Now that we have built several version of the corridor, we will now build the top surface representation of the Corridor. We will be working on the latest version of the Intersection design. There is a file called **Intersection Example 3 - Complete.dwg** which you can start from.

33. Open the drawing called **Intersection Example 3 – Complete.dwg** , select the Corridor, and build a Corridor Surface from the Links coded Top. You should create a new Surface Style called **Corridor Surface** for displaying the Corridor. Set the Contour interval to 0.5' and 2.5' for the minor and major respectively, and set the display to the Major and Minor contours.
34. The next step requires building a Corridor boundary. The following figure illustrates the order which to proceed with the Border:



There will be 7 feature lines which make up the Border. Once you have selected all the feature lines you will need to go into the Border properties dialog and set the appropriate stations. The following dialog has each of the feature lines, starting from the Top right and working counter-clockwise around the corridor. You should refer to the following dialog in assigning the stations for each of the feature lines.

Feature Line	Baseline	Reverse Direction	Start Point	End Point
Daylight(0)	Barberry Tree Cove CL	<input type="checkbox"/>	42+00.00'	44+00.00'
Daylight(1)	Barberry Tree Cove CL	<input checked="" type="checkbox"/>	43+62.10'	44+00.00'
Daylight(0)	Flowering Spring Trail CL	<input type="checkbox"/>	0+00.00'	0+34.00'
Daylight(0)	Flowering Spring Trail CL	<input type="checkbox"/>	0+34.00'	1+00.00'
Daylight(1)	Flowering Spring Trail CL	<input checked="" type="checkbox"/>	0+34.00'	1+00.00'
Daylight(0)	Flowering Spring Trail CL	<input checked="" type="checkbox"/>	0+00.00'	0+34.00'
Daylight(1)	Barberry Tree Cove CL	<input checked="" type="checkbox"/>	42+00.00'	42+94.10'

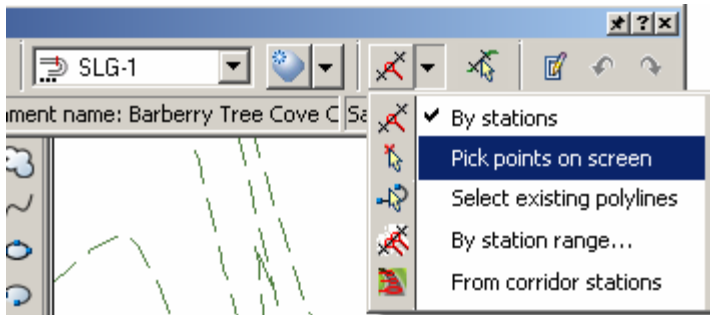


This can be tedious so you will find a drawing with a Corridor surface and a Boundary created. The drawing is called **Intersection Example 3 – Complete with Surface.dwg**.

## Analyzing the Corridor Surface

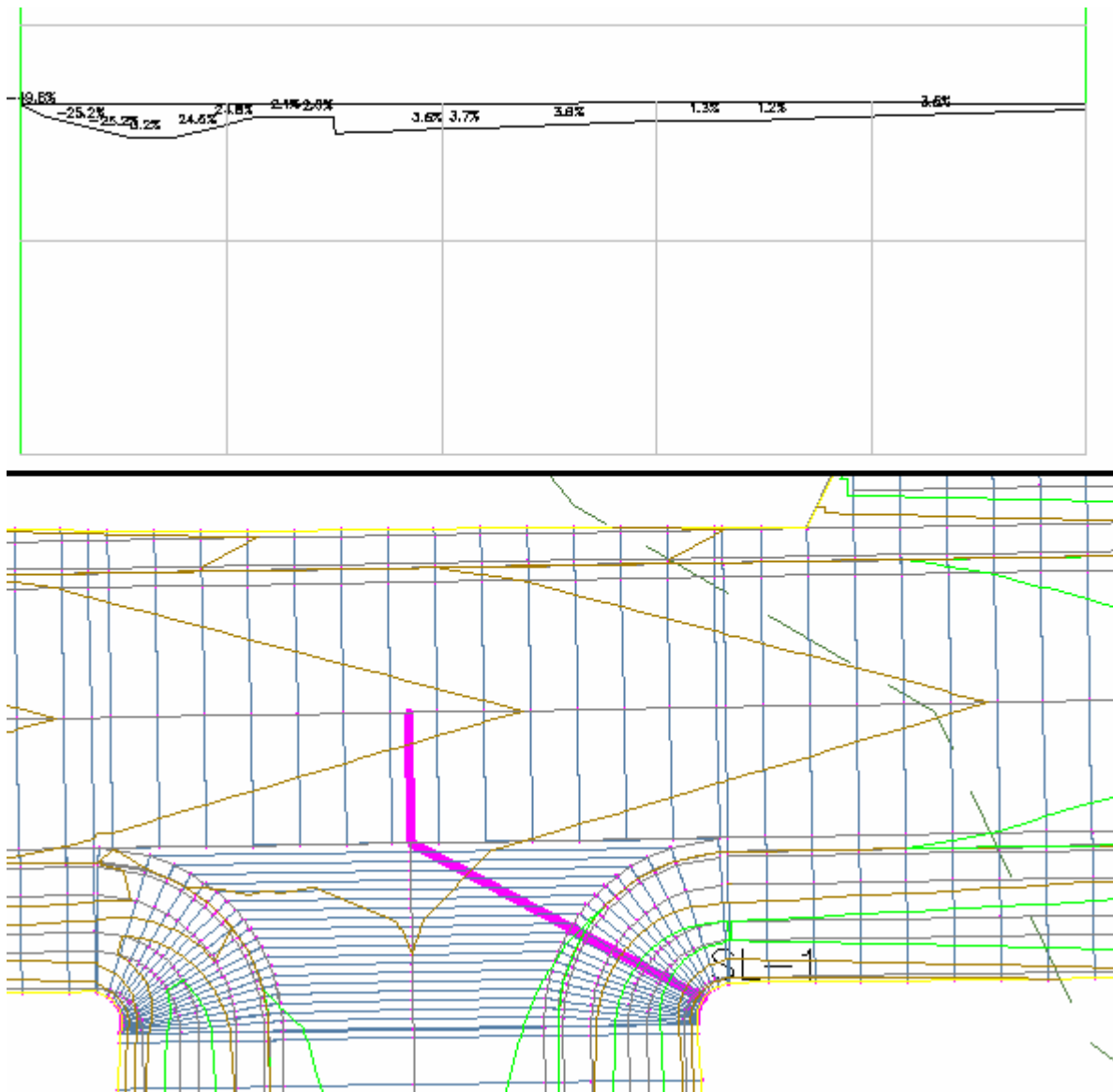
Now that we have a well formed boundary on the corridor surface, sections can be cut in arbitrary locations within the Corridor. We are going to create sections which run through the intersection in order to verify correct slope characteristics. The surface analysis tools will also be used to present information useful in analyzing the surface.

35. Open the Drawing called **Intersection Example 3 – Surface Analysis.dwg**. You will notice that there are 2 viewports in this drawing – we are going to use the top viewport for the section view. Create a sample which starts at the intersection of the 2 alignments and then passes through the curb return of the intersection. Select the command **Sections->Create Sample Lines...** from the menu and select the main alignment, **Barberry Tree Cove CL alignment**. One the Create Sample Line Group dialog, accept the defaults for the styles and the sections to Sample. One the sample line group Toolbar, select the option to create the sample line by Picking Points. This is shown in the following illustration:



Select the intersection point of the 2 alignments, then select the edge of travel way for the main alignment (perpendicular to the alignment), and then select a point in the curb return region. Select Enter and you will now have a sample line.

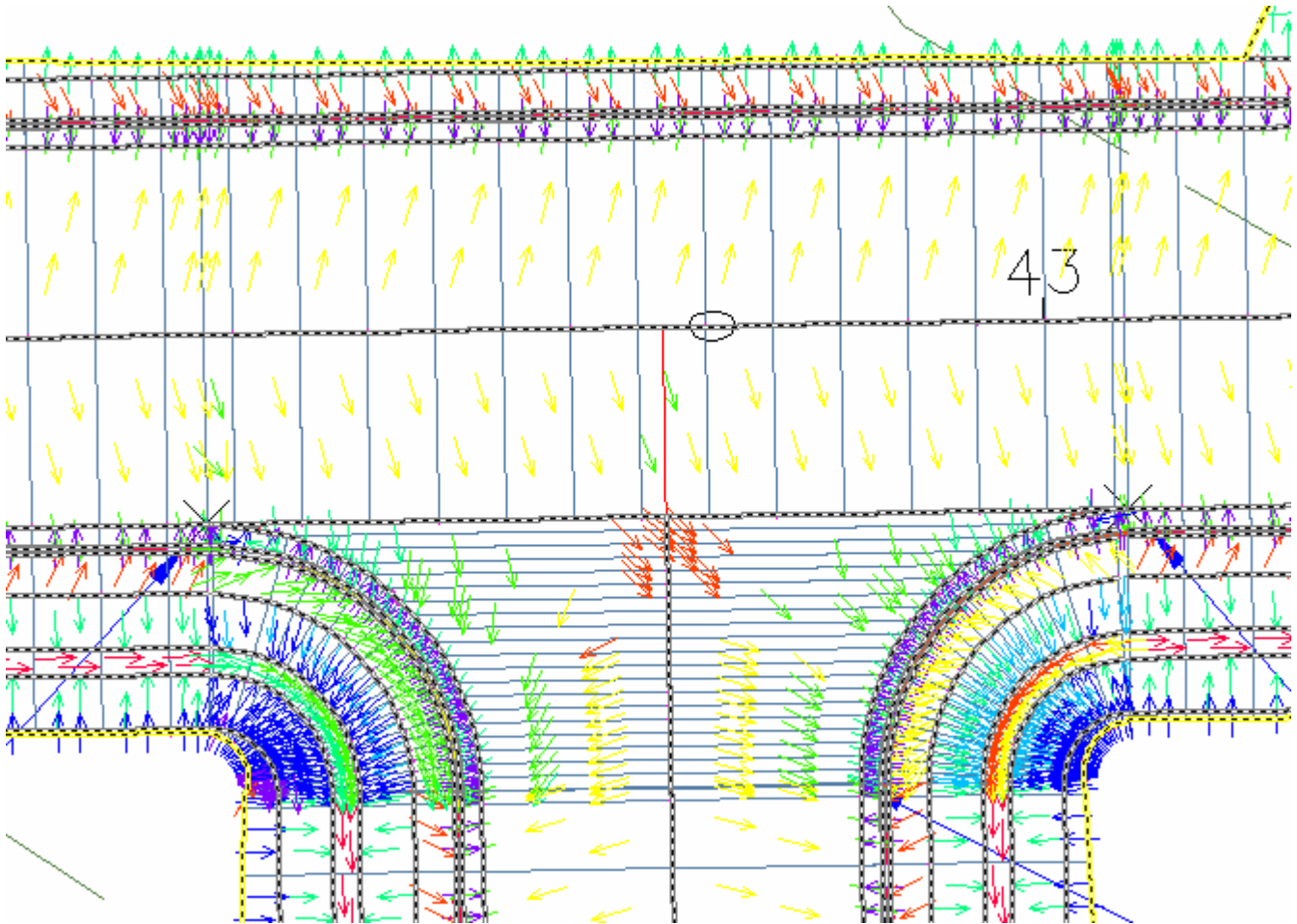
36. Lastly, create a Section View by selecting the command **Section -> Create View...** from the menu. Drop the section view into the top viewport. You should have a section view in the top viewport and the intersection design in the bottom viewport. The view should look like the following (the Sample line style has been changed for illustration):



Now you can make edits to sample line and make decisions regarding the slope analysis and locate the catch basins.

The surface style is also a useful way to analyze the characteristics of the top surface. For example, we can set the surface to display slope arrows and this is a visual way of analyzing the surface.

37. Close the previous drawing and re-open **Intersection Example 3 – Surface Analysis.dwg**. Reset the view to a single viewport and only display the intersection surface. Select the style called **Corridor Surface** and change the display from Contours to Slope Arrows. You can change the appearance of the slope arrows by selecting the style again – change the length of the arrows to 3' and also change the arrow head to open.



This gives a good picture of the slope directions and magnitudes.

## Conclusion

Autodesk Civil 3D can effectively be used to model intersections. This lab explored different techniques for modeling typical intersections. The type of engineering design will dictate the actual method.

