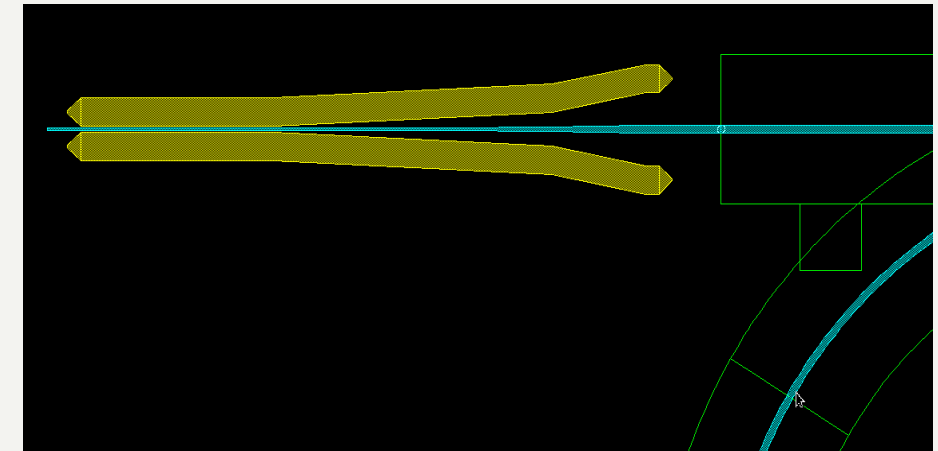
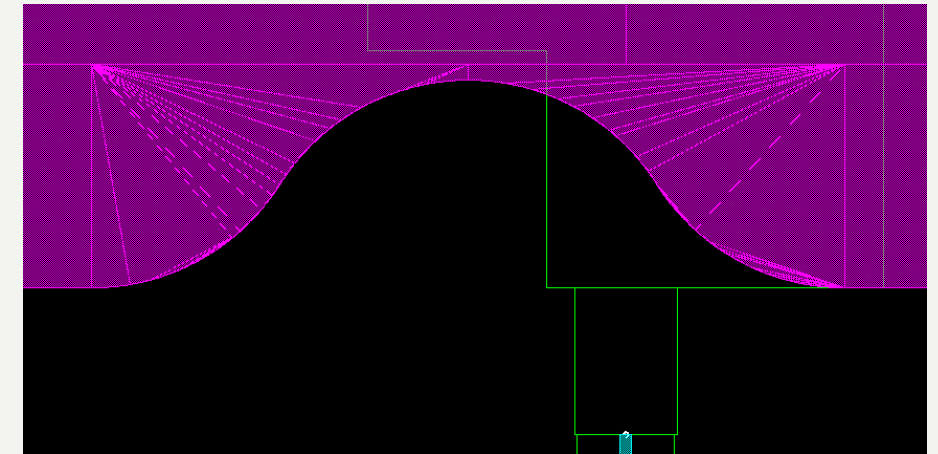
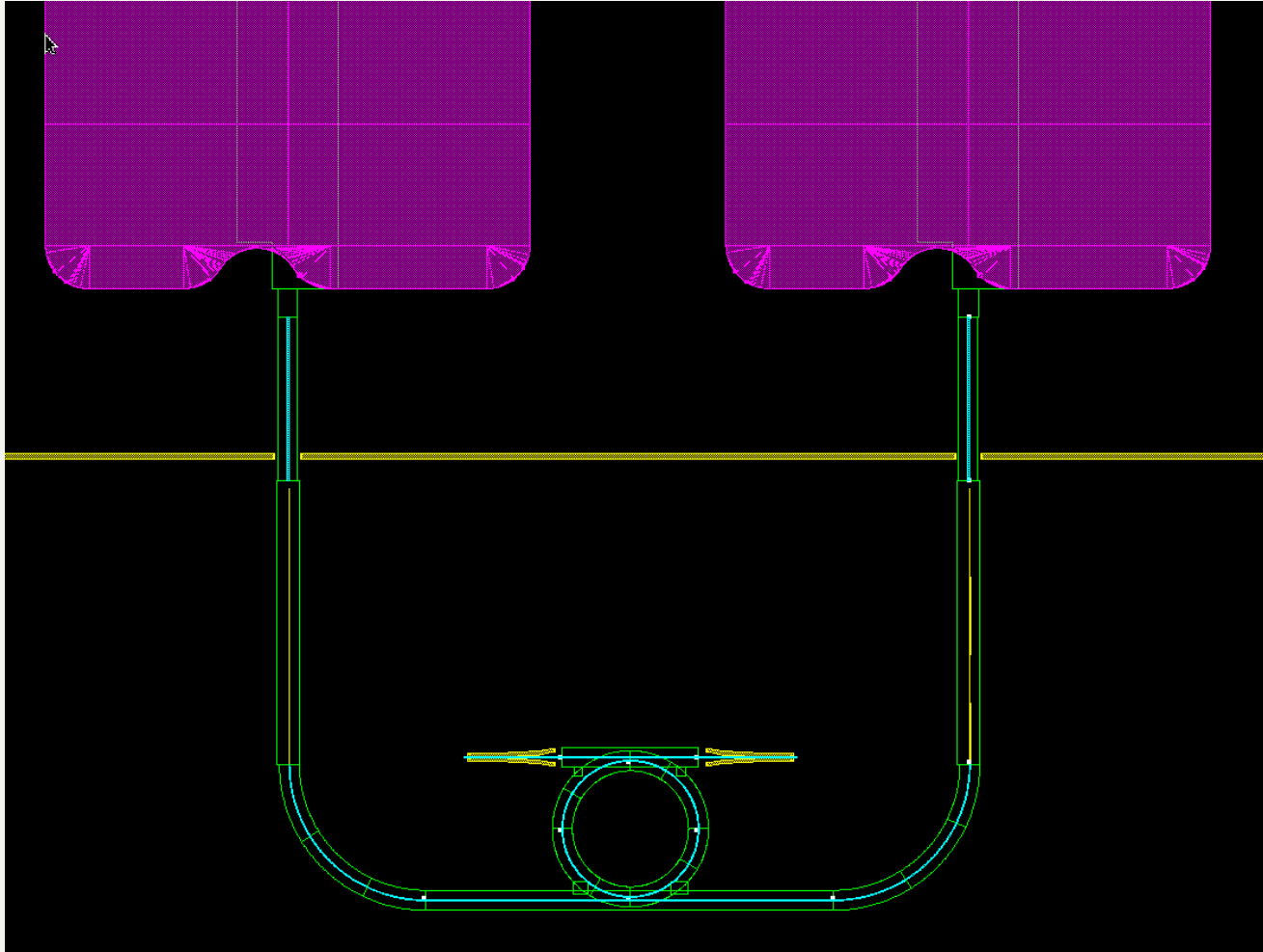


# Radius Curvatures – SVRF Code

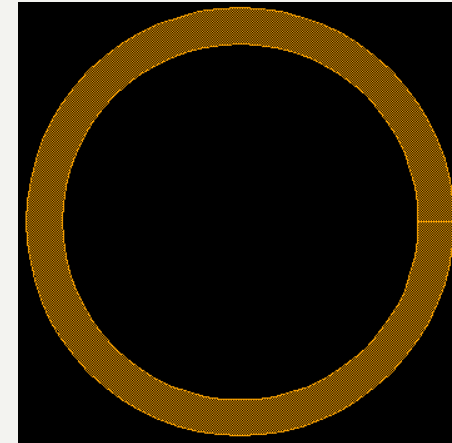
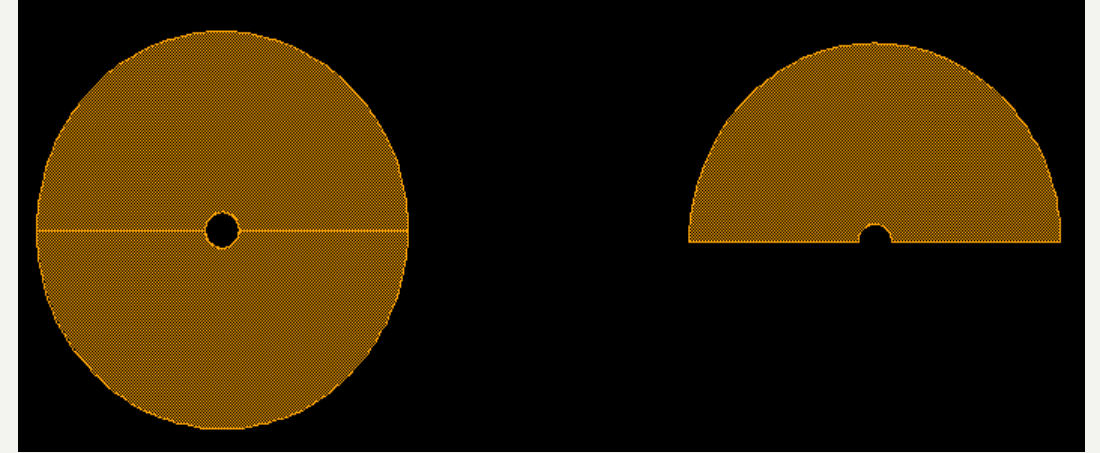
*How to Verify Photonic Curvilinear Shapes*

# 1. Introduce Various Curvatures / FreeForm

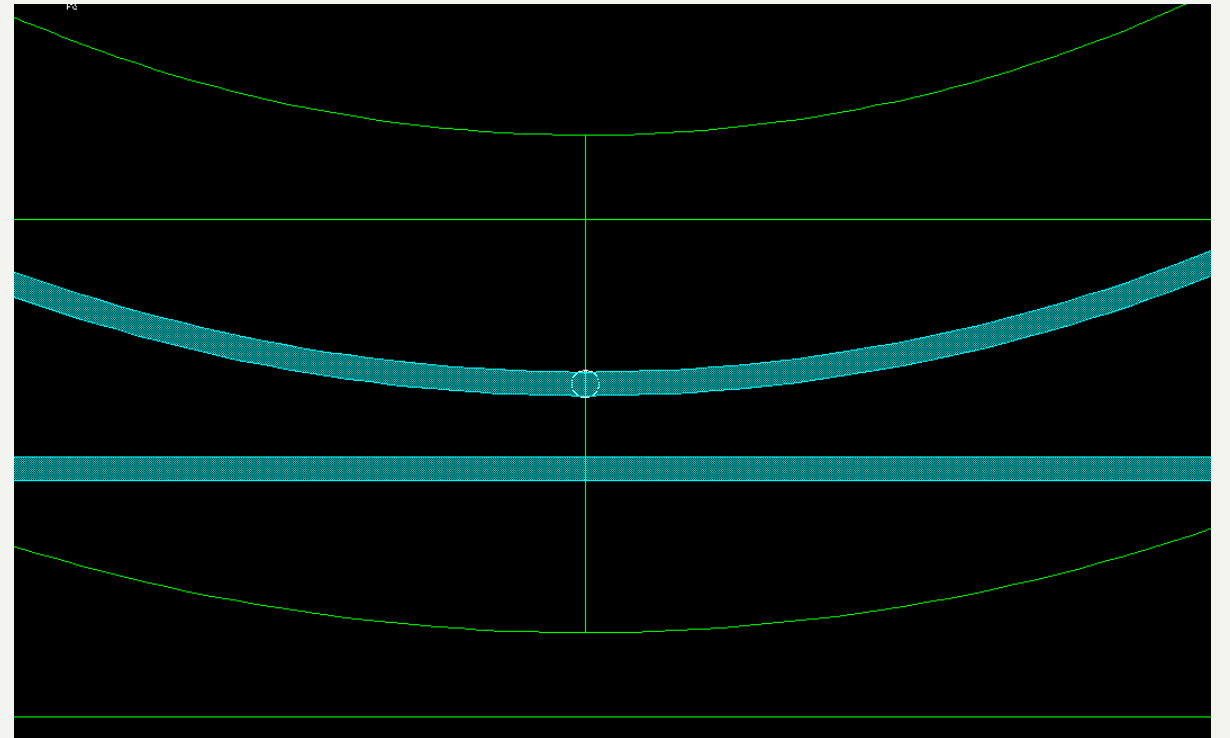
# 1. Introduce Various Curvatures / FreeForm



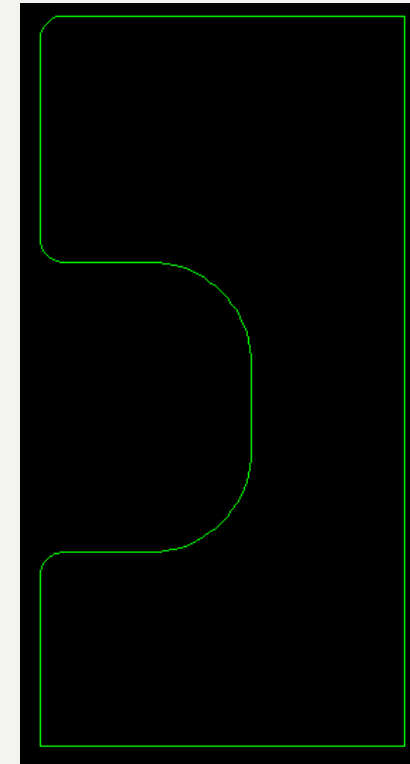
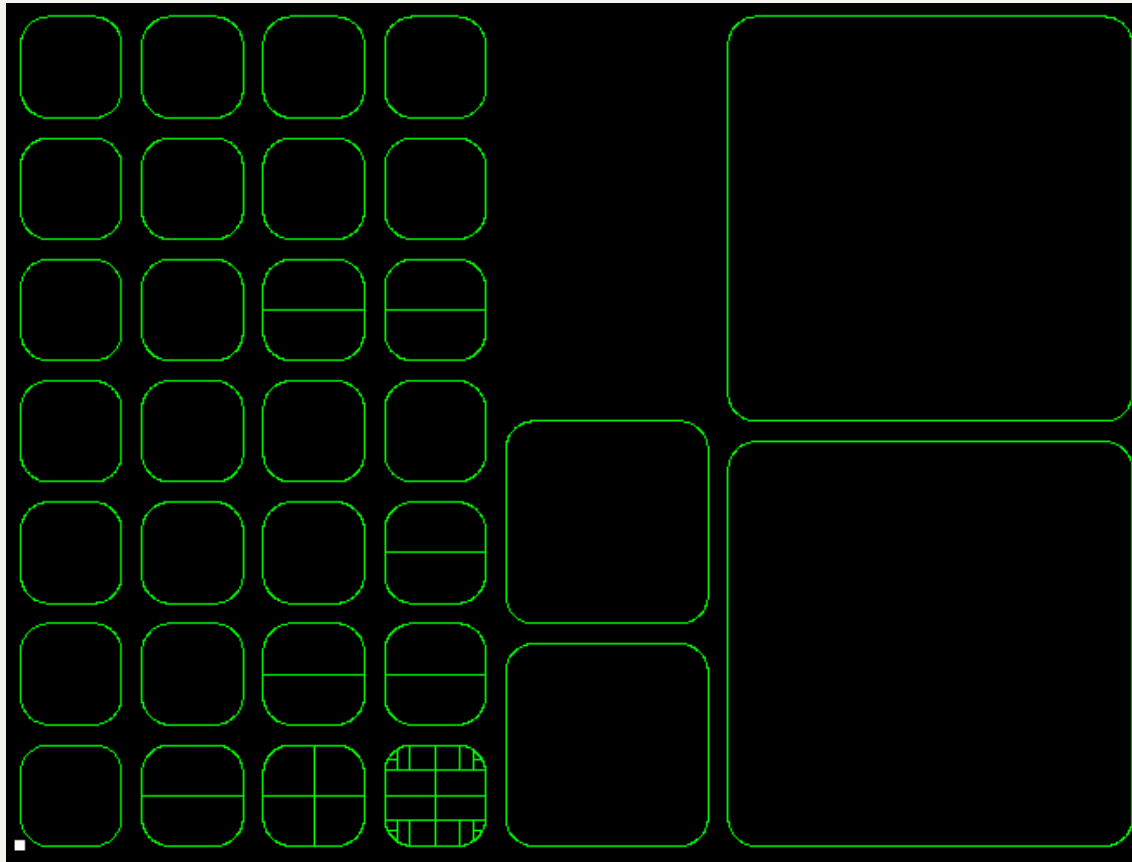
# 1. Introduce Various Curvatures / FreeForm



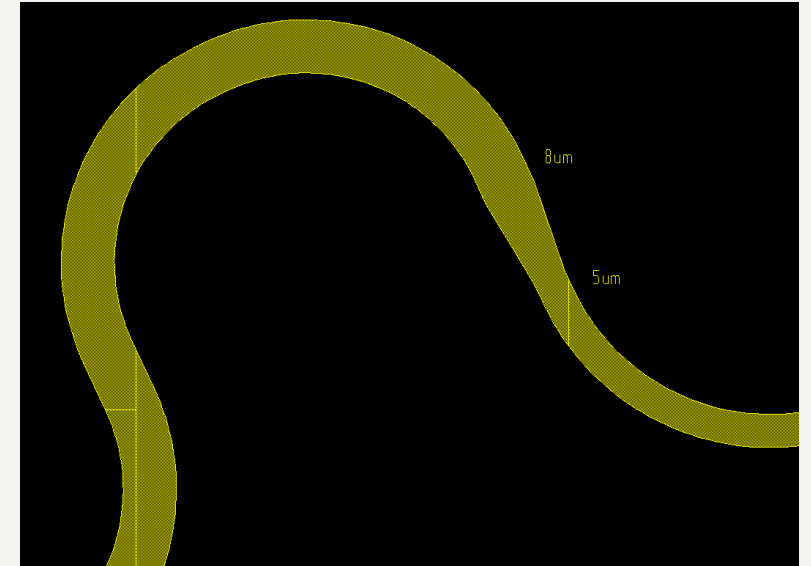
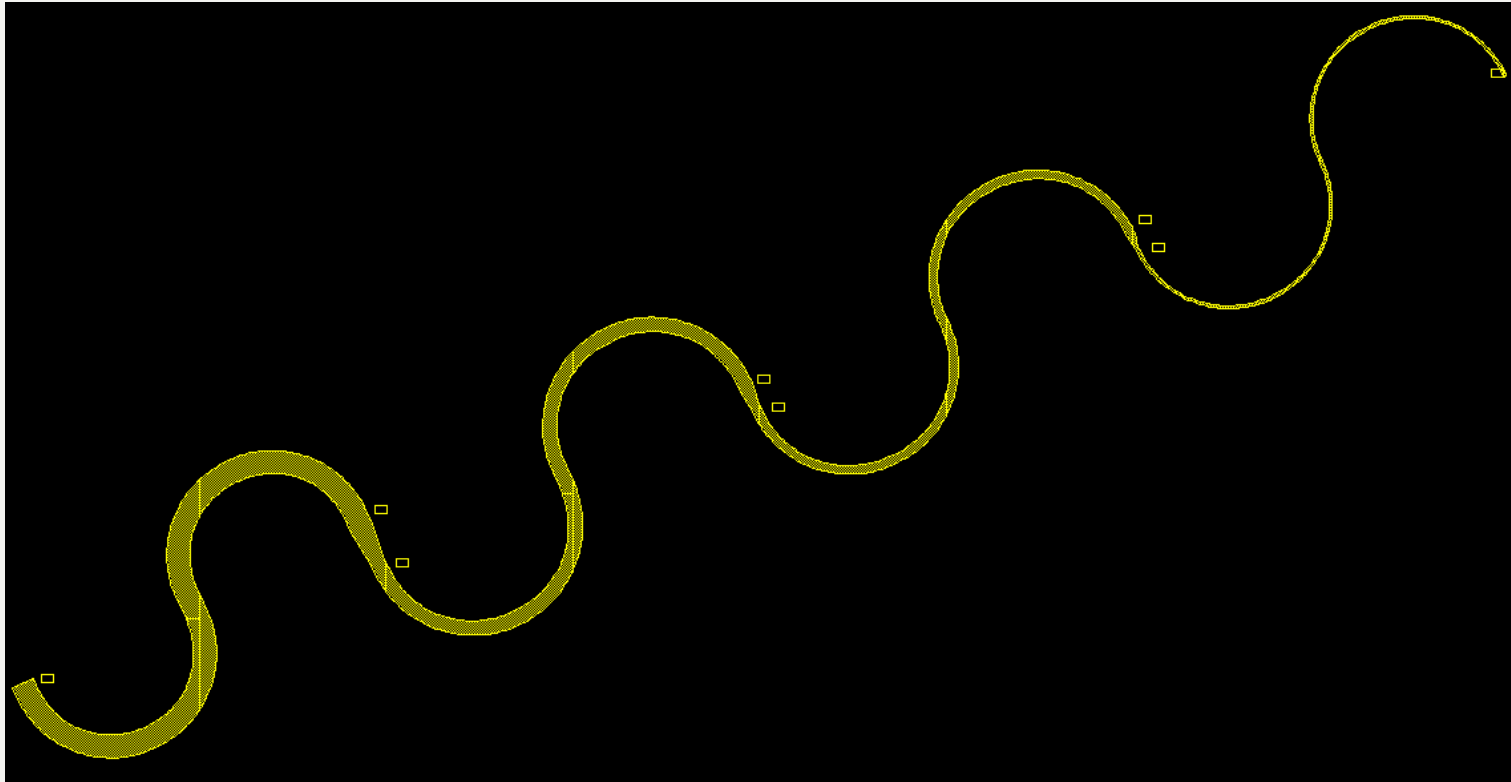
# 1. Introduce Various Curvatures / FreeForm



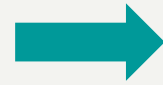
# 1. Introduce Various Curvatures / FreeForm



# 1. Introduce Various Curvatures / FreeForm



# 1. Issues with Curvatures / FreeForm

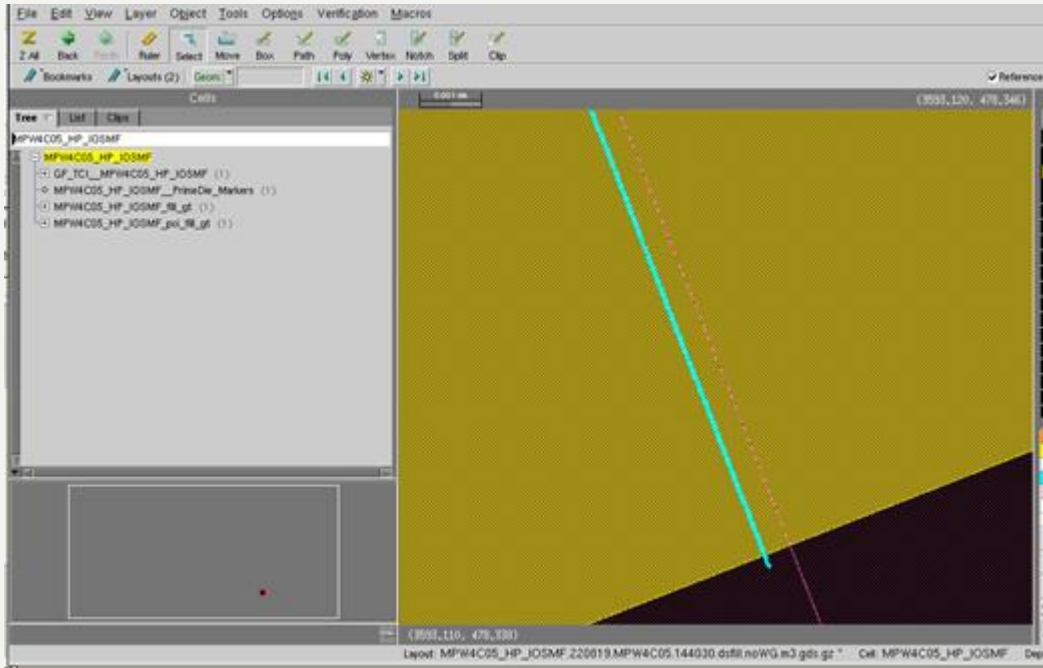


some slots and stitched outputs are all on output edge line

EXT (SIZE M1 BY 0.25) < 1.2 ABUT < 90 OPPOSITE REGION

BW\_wo\_gap = (BW\_gen1 OR ([EXT BW\\_gen1 < 1.2 ABUT < 90 OPPOSITE REGION](#) )) OR (EXT (BW\_gen1 OR ([EXT BW\\_gen1 < 1.2 ABUT < 90 OPPOSITE REGION](#))) < 1.2 ABUT < 90 OPPOSITE REGION)

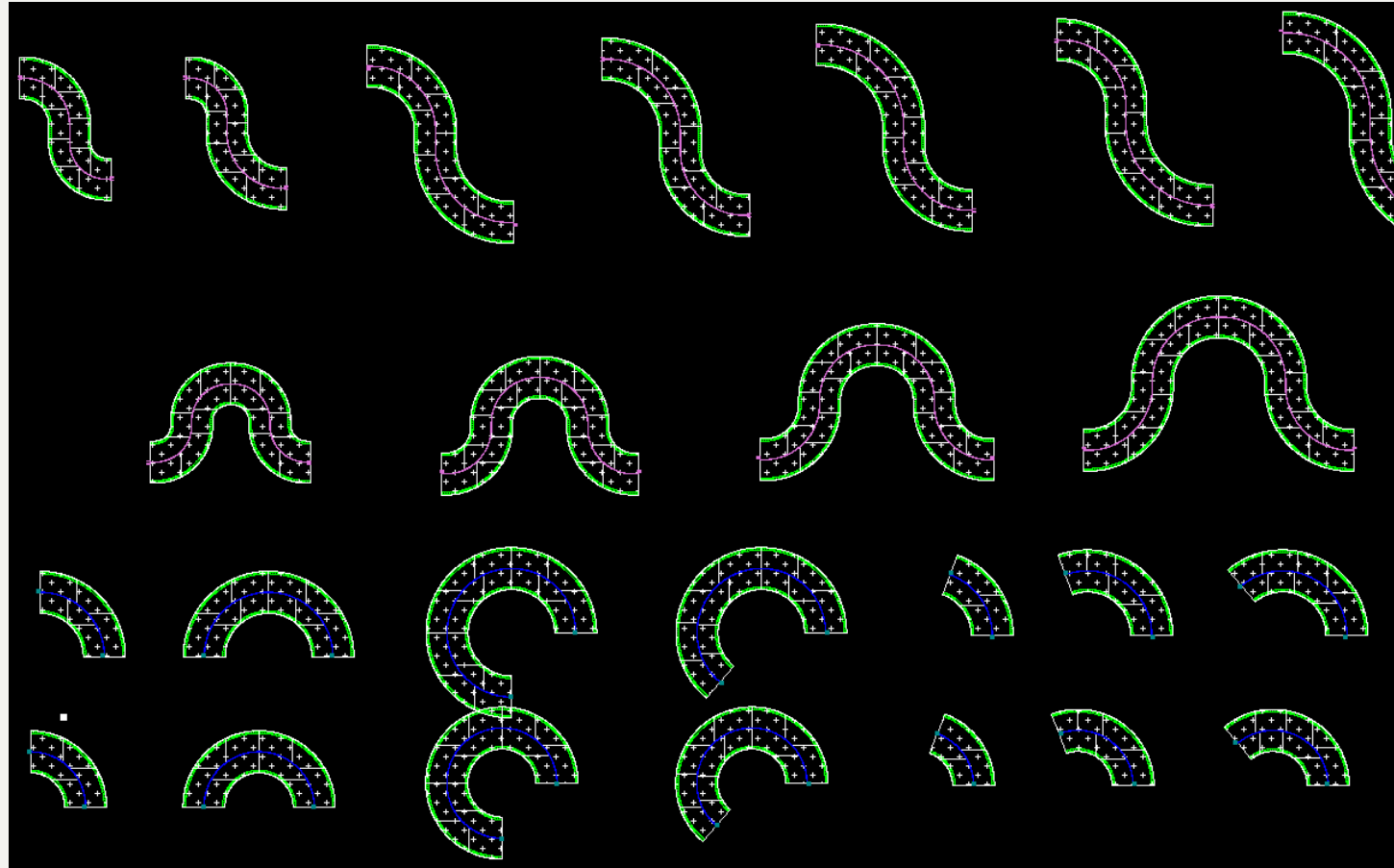
# 1. Issues with Curvatures / FreeForm



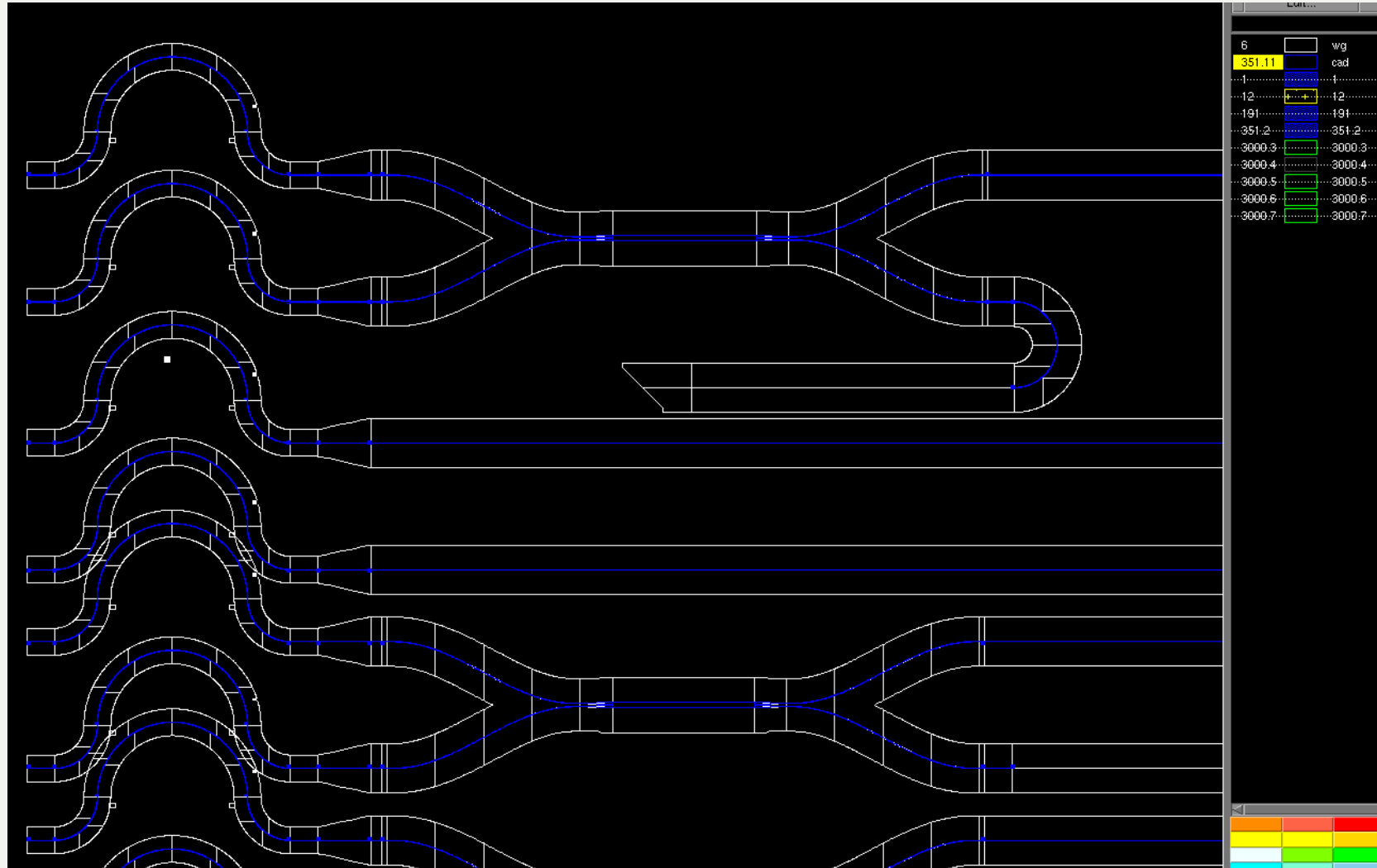
INT (RX NOT PHOTON) < 0.12 ABUT<90 SINGULAR

When doing "A NOT B", when B has skew edges such as ring, it left some slivers along boundary of two polygons formed the ring

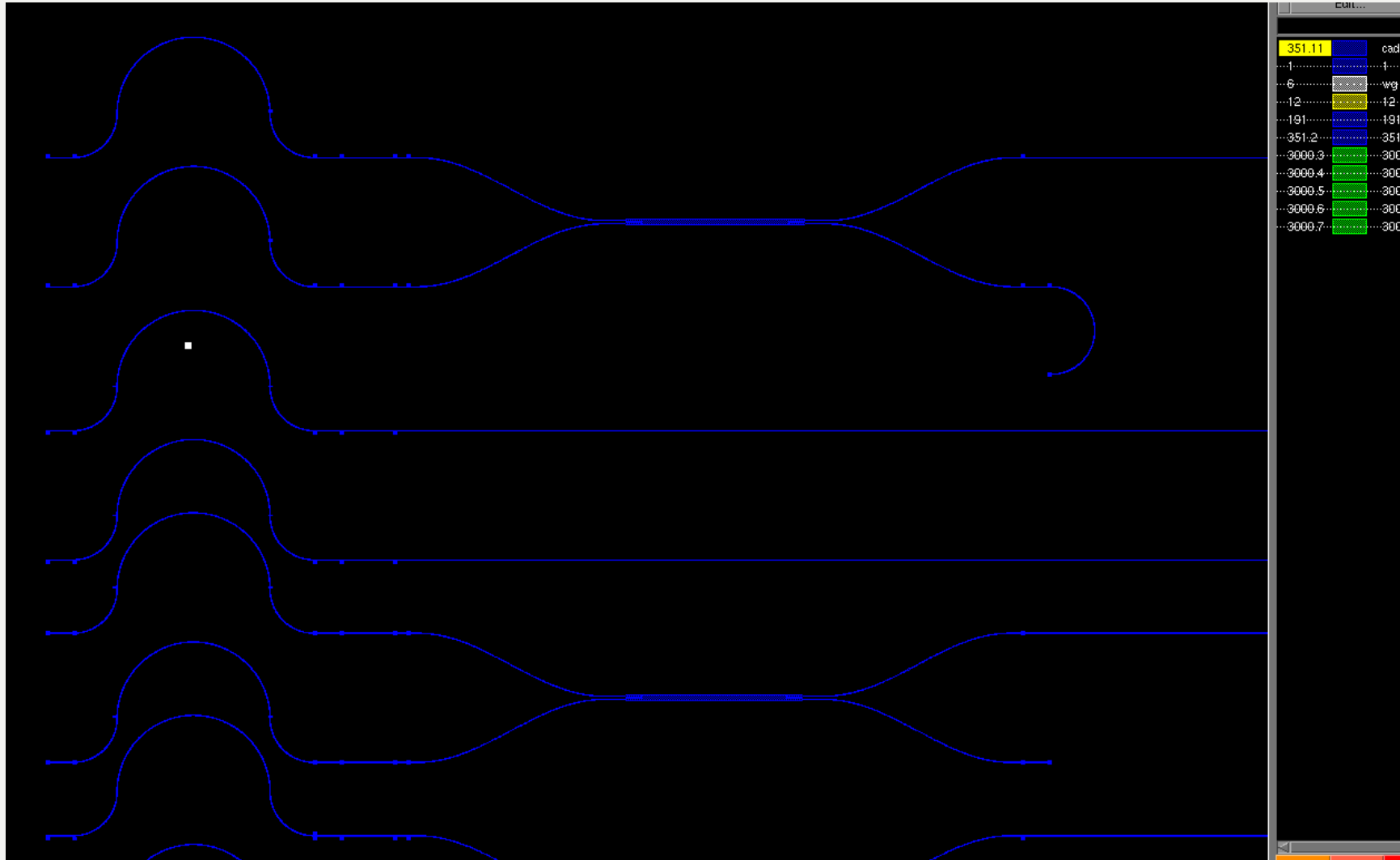
# 1. Introduce Various Curvatures / FreeForm



# 2. Design with Radius Curvatures

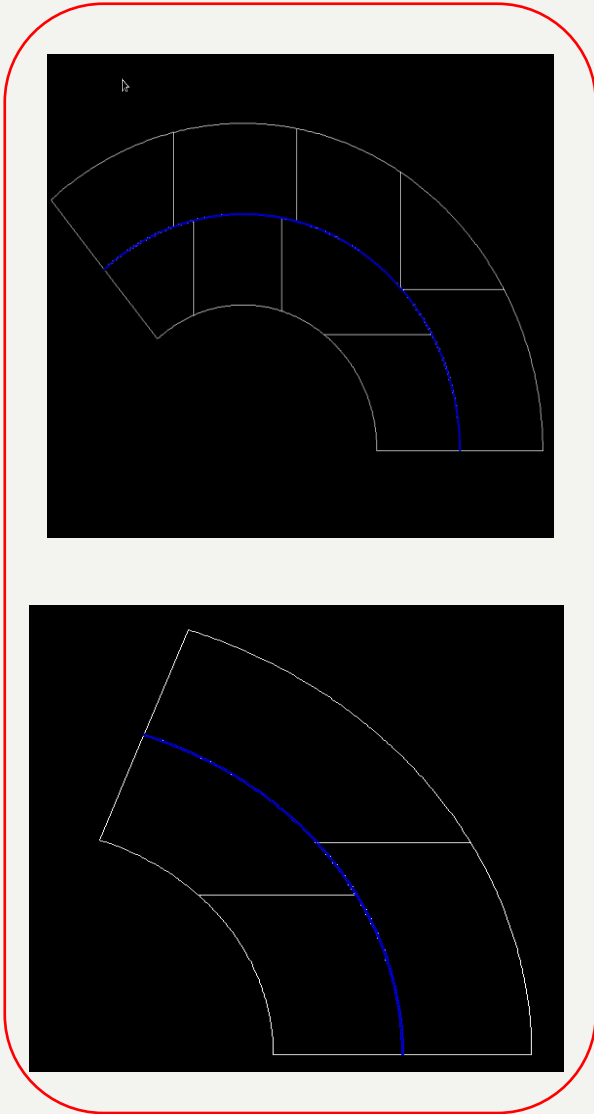


# 2. Design with Radius Curvatures

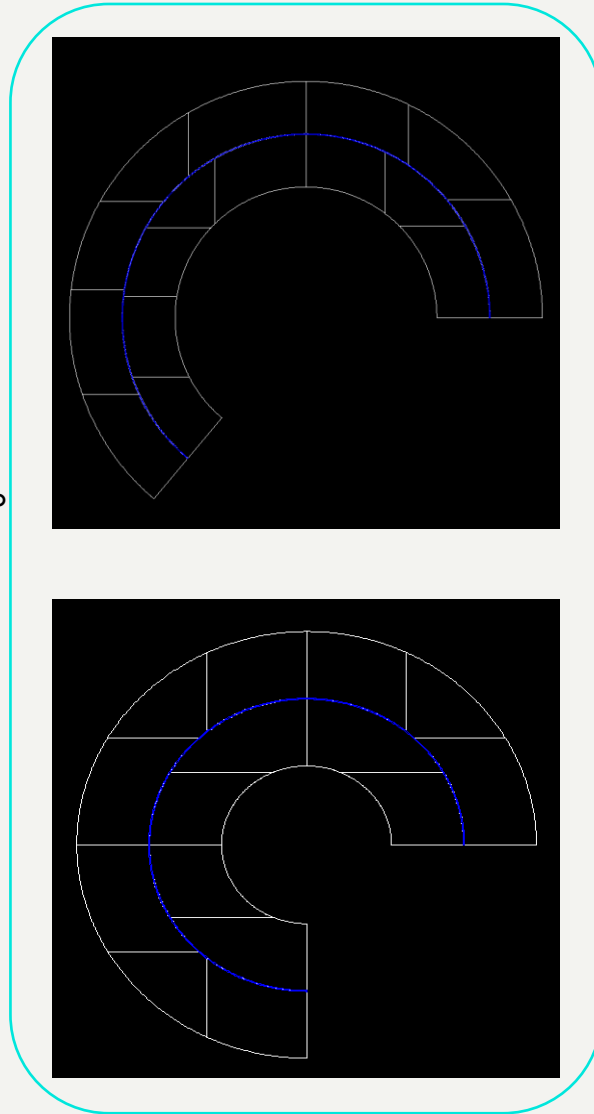


# 2. Design with Radius Curvatures

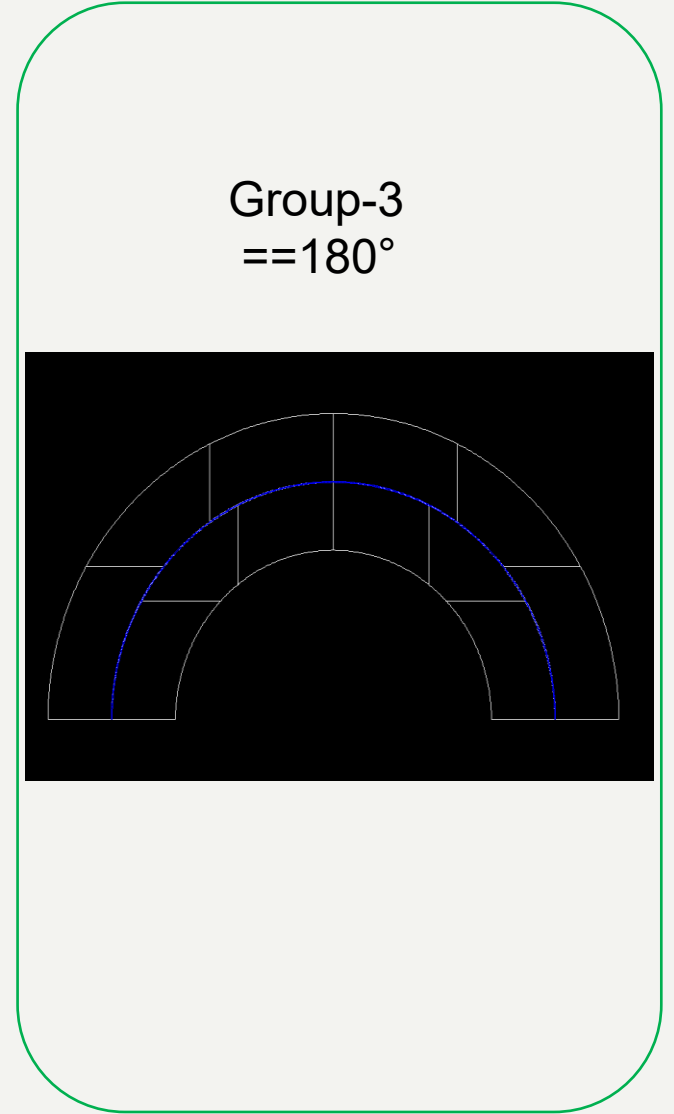
Group-1  
 $0 > < 180^\circ$



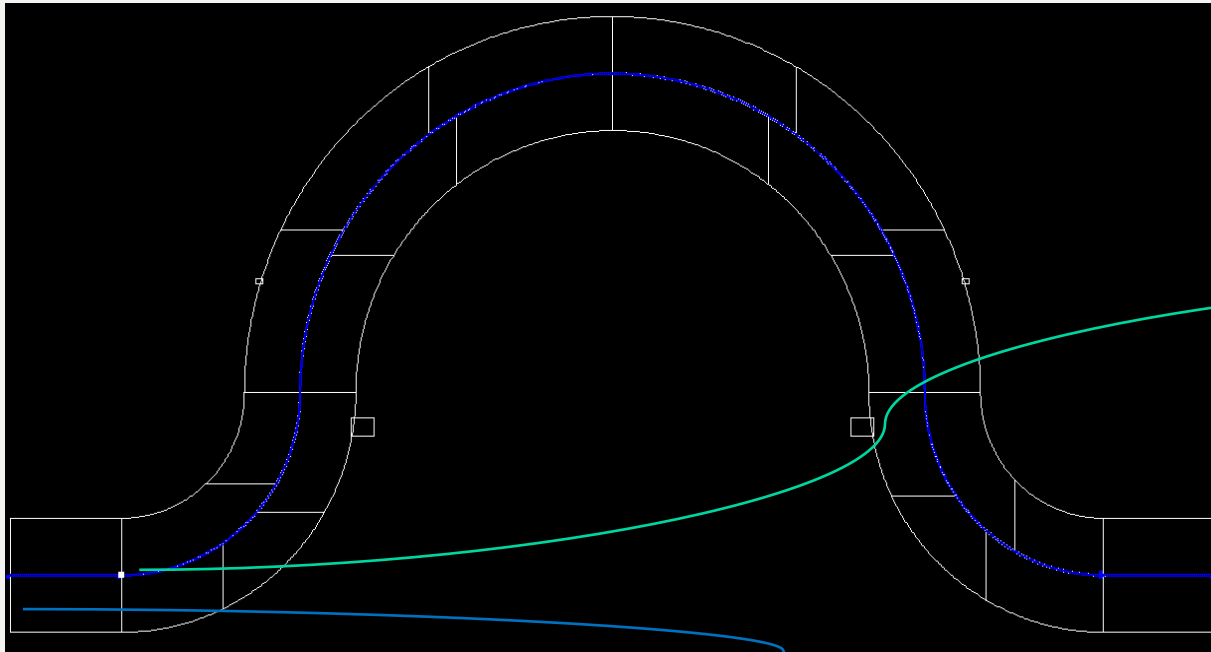
Group-2  
 $180 > < 360^\circ$



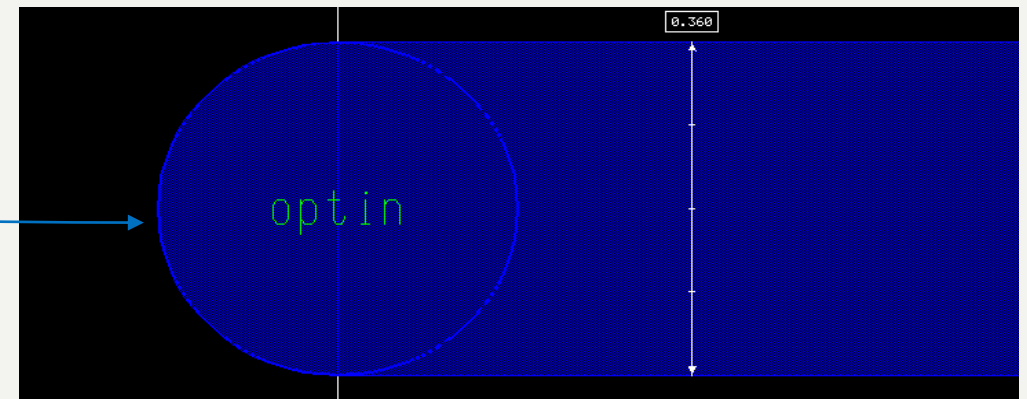
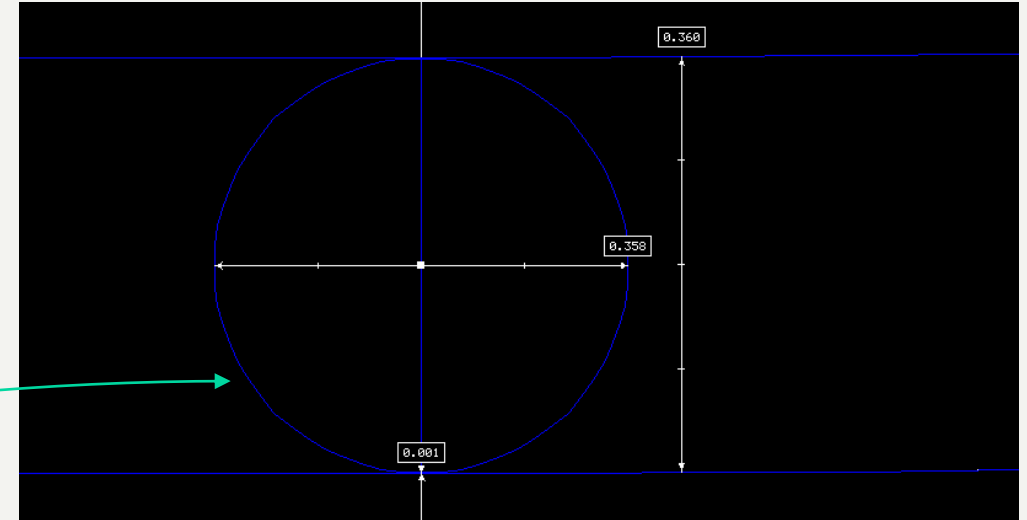
Group-3  
 $= 180^\circ$



# 2. Design with Radius Curvatures



Zoom



# Sample Rules

```
//Check angles less than 180 degrees 0 > < 180
INT_ends = INT P_ends < maximum_radius PERpendicular ALSO OBTUSE ALSO
INT_angle = DFM PROPERTY INT_ends P_ends OR P_ends_exp OVERLAP ABUT ALSO SINGULAR MULTI
[radians = SUM(ANGLE(INT_ends))] > 0
[degrees = PROPERTY_REF(radians)*180/PI() ]
[width = MAX(LENGTH(P_ends))]
[distance = EW(INT_ends) ] > minimum_grid*3
[Radius = (AREA(OR_P_ends_exp))/(PROPERTY_REF(width)*PROPERTY_REF(radians))]
INT_angle {
DFM COPY INT_angle DFM RDB INT_angle "dfm.rdb" all cells checkname "%_l_" }
```

Group-1

0 > < 180°

```
//Check between 180 degrees and 360 degrees
EXT_ends = EXT P_ends NOTCH PERpendicular ALSO OBTUSE ALSO < maximum_radius
EXT_angle = DFM PROPERTY EXT_ends P_ends OR P_ends_exp OVERLAP ABUT ALSO SINGULAR MULTI
[radians = SUM(ANGLE(EXT_ends))] > 0
[degrees = 360 -(PROPERTY_REF(radians)*180/PI()) ]
[width = MAX(LENGTH(P_ends))]
[distance = EW(EXT_ends) + PROPERTY_REF(width) ] > minimum_grid*3
[distance_edges = EW(EXT_ends) ] > minimum_grid*3
[COS theta formular = COS(PROPERTY_REF(radians)/2)]
[Radius = (AREA(OR_P_ends_exp))/(PROPERTY_REF(width)*PROPERTY_REF(radians))]
EXT_angle {
DFM COPY EXT_angle DFM RDB EXT_angle "dfm.rdb" all cells checkname "%_l_" }
```

Group-2

180 > < 360°

```
//Check 180 degrees using EXT_ends and INT_ends above
NOT_TOUCH_EDGE = P_ends NOT TOUCH_EDGE (DFM COPY EXT_ends INT_ends EDGE)
EXPAND_EDGE_NOT_TOUCH_EDGE = EXPAND EDGE NOT_TOUCH_EDGE INSIDE BY minimum_grid
EXPAND_EDGE_NOT_TOUCH_EDGE {DFM COPY EXPAND_EDGE_NOT_TOUCH_EDGE}

CONNECT EXPAND_EDGE_NOT_TOUCH_EDGE OR P_ends_exp

EXT_ends_180 = EXT EXPAND_EDGE_NOT_TOUCH_EDGE < maximum_radius CONNECTED
EXT_ENDS_180 {DFM COPY EXT_ends_180}

EXT_angle_180 = DFM PROPERTY EXT_ends_180 P_ends OR P_ends_exp OVERLAP ABUT ALSO SINGULAR MULTI
[radians = SUM(ANGLE(EXT_ends_180))] <= 0
[degrees = 180 -(PROPERTY_REF(radians)*180/PI())]
[width = MAX(LENGTH(P_ends))]
[distance = EW(EXT_ends_180) + PROPERTY_REF(width) ] > minimum_grid*3
[distance_edges = EW(EXT_ends_180) ] > minimum_grid*3
[Radius = PROPERTY_REF(distance)/2 ]
EXT_angle_180 {
DFM COPY EXT_angle_180 DFM RDB EXT_angle_180 "dfm.rdb" all cells checkname "%_l_" }
```

Group-3

== 180°

# Sample Rules

```
//Check angles less than 180 degrees 0 > < 180
INT_ends = INT P_ends < maximum_radius PERpendicular ALSO OBTUSE ALSO
INT_angle = DFM PROPERTY INT_ends P_ends OR P_ends_exp OVERLAP ABUT ALSO SINGULAR MULTI
[radians = SUM(ANGLE(INT_ends))] > 0
[degrees = PROPERTY_REF(radians)*180/PI() ]
[width = MAX(LENGTH(P_ends))]
[distance = EW(INT_ends) ] > minimum_grid*3
[Radius = (AREA(OR_P_ends_exp))/(PROPERTY_REF(width)*PROPERTY_REF(radians))]
INT_angle {
DFM COPY INT_angle DFM RDB INT_angle "dfm.rdb" all cells checkname "%_l_" }
```

Group-1  
0 > < 180°

dfm.rdb x drc.db

Filter: Show All Mentor2, 799 Results (in 5 of 6 Checks)

Check / Cell	Results
Check P_length_Adj_prop::<2>_1	419
Check near_circle	229
Check EXT_angle_180	69
Check out_corner	54
Check INT_angle	28
Check EXT_angle	0

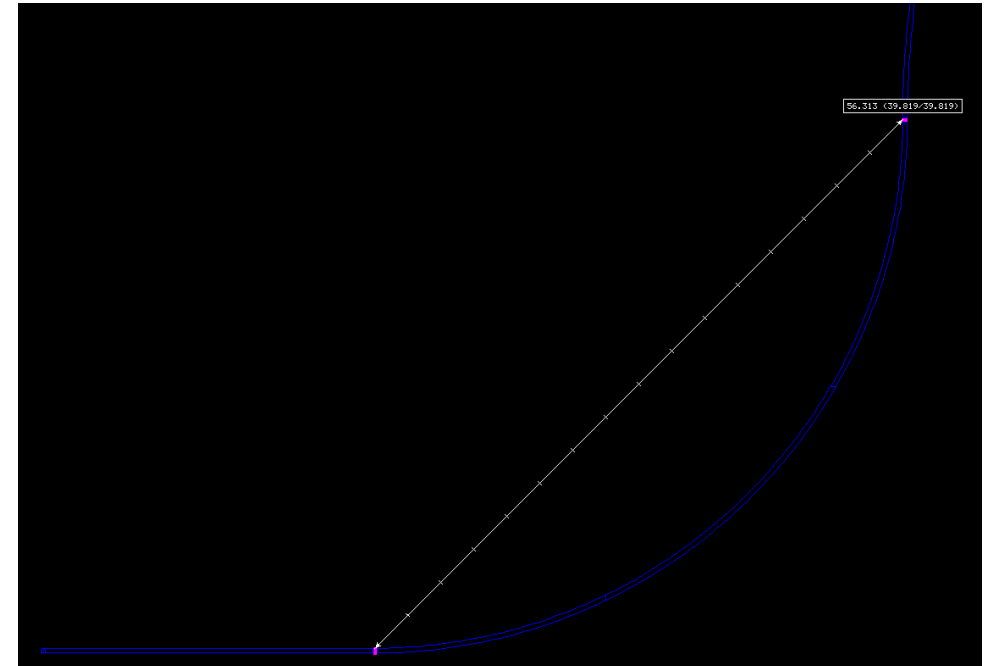
	degrees	distance	radians	Radius	width	Vertices
1	90	56.3117	1.57080	40.0051	0.360000	4
2	90	56.3117	1.57080	40.0051	0.360000	4
3	90	56.3117	1.57080	40.0051	0.360000	4
4	90	56.3117	1.57080	40.0051	0.360000	4
5	90	56.3117	1.57080	40.0051	0.360000	4
6	90	56.3117	1.57080	40.0051	0.360000	4
7	90	56.3117	1.57080	40.0051	0.360000	4
8	90	56.3117	1.57080	40.0051	0.360000	4
9	90	56.3117	1.57080	40.0051	0.360000	4

1) Result ID 718 Check INT\_angle, Cell Mentor2: 2 Edges

width 0.36  
radians 1.5708  
degrees 90  
distance 56.3117  
Radius 40.0051

2 Edges. coordinates in cell Mentor2

```
{ -65.7619 -1144.2429 } { -65.4019 -1144.2429 }
{ -105.5801 -1184.4211 } { -105.5801 -1184.0614 }
```



# Sample Rules

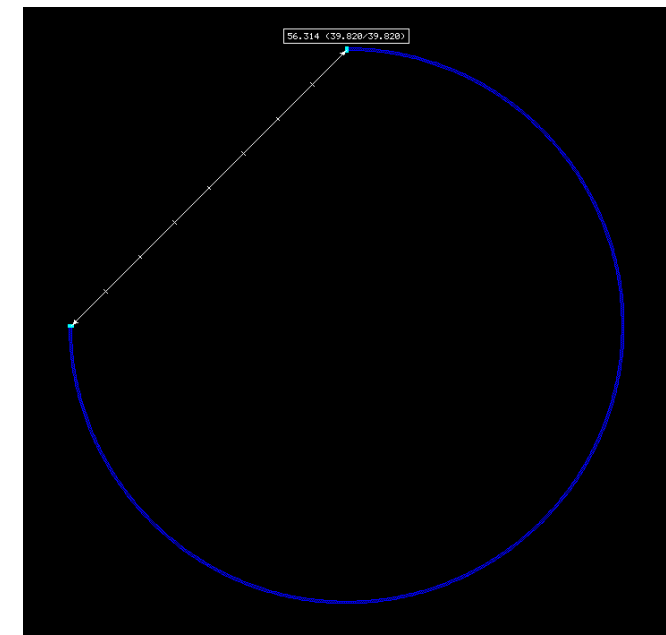
```
//Check between 180 degrees and 360 degrees
EXT_ends = EXT P_ends NOTCH PERpendicular ALSO OBTUSE ALSO < maximum_radius
EXT_angle = DFM PROPERTY EXT_ends P_ends OR P_ends_exp OVERLAP ABUT ALSO SINGULAR MULTI
[radians = SUM(ANGLE(EXT_ends))] > 0
[degrees = 360 - (PROPERTY_REF(radians)*180/PI()) ]
[width = MAX(LENGTH(P_ends))]
[distance_centre = EW(EXT_ends) + PROPERTY_REF(width)] > minimum_grid*3
[distance = EW(EXT_ends) ] > minimum_grid*3
[COS_theta_formular = COS(PROPERTY_REF(radians)/2)]
[Radius = (AREA(OR_P_ends_exp))/(PROPERTY_REF(width)*PROPERTY_REF(radians))]
EXT_angle {
DFM COPY EXT_angle DFM RDB EXT_angle "dfm.rdb" all cells checkname "%_l_" }
```

Group-2  
>180° < 360°

dfm.rdb x drc.db

Filter: Show All TM\_Filter\_NarrowSlab, 47 Results (in 6 of 6 Checks)

Check / Cell	Results	degrees	radians	Radius	width	distance	distance_edges	COS_theta_formular	Vertices	
Check out_corner	16	1	270	1.57080	120.118	0.359700	56.6742	56.3145	0.707107	4
Check P_length_Adj_prop::<2>_1	12									
Check INT_angle	9									
Check near_circle	6									
Check EXT_angle_180	3									
Check EXT_angle	1									



# Sample Rules

```
//Check 180 degrees using EXT_ends and INT_ends above
NOT_TOUCH_EDGE = P_ends NOT TOUCH EDGE (DFM COPY EXT_ends INT_ends EDGE)
EXPAND_EDGE_NOT_TOUCH_EDGE = EXPAND EDGE NOT_TOUCH_EDGE INSIDE BY minimum_grid
EXPAND_EDGE_NOT_TOUCH_EDGE {DFM COPY EXPAND_EDGE_NOT_TOUCH_EDGE}

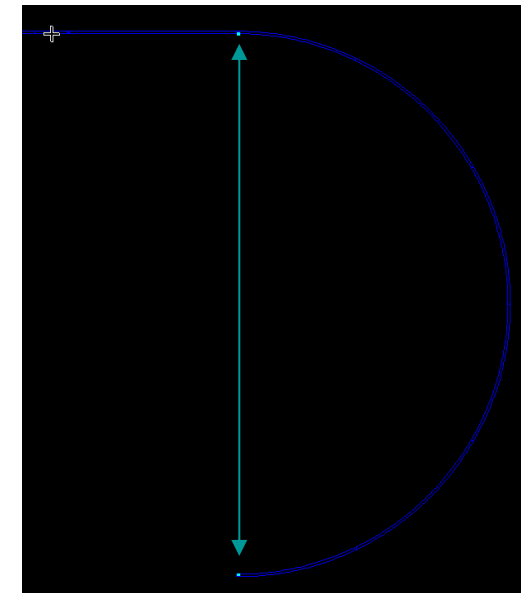
CONNECT EXPAND_EDGE_NOT_TOUCH_EDGE OR_P_ends_exp

EXT_ends_180 = EXT EXPAND_EDGE_NOT_TOUCH_EDGE < maximum_radius CONNECTED
EXT_ENDS_180 {DFM COPY EXT_ends_180}

EXT_angle_180 = DFM PROPERTY EXT_ends_180 P_ends OR_P_ends_exp OVERLAP ABUT ALSO SINGULAR MULTI
[radians = SUM(ANGLE(EXT_ends_180))] <= 0
[degrees = 180 - (PROPERTY_REF(radians)*180/PI())]
[width = MAX(LENGTH(P_ends))]
[distance = EW(EXT_ends_180) + PROPERTY_REF(width) ] > minimum_grid*3
[Radius = PROPERTY_REF(distance)/2 ]

EXT_angle_180 {
DFM COPY EXT_angle_180 DFM RDB EXT_angle_180 "dfm.rdb" all cells checkname "%_l_" }
```

Group-3  
==180°



dfm.rdb | drc.db

Filter: Show All | Mentor2, 799 Results (in 5 of 6 Checks)

Check / Cell	Results /
Check P_length_Adj_prop::<2>_1	419
Check near_circle	229
Check EXT_angle_180	69
Check out_corner	54
Check INT_angle	28
Check EXT_angle	0

degrees	distance	radians	Radius	width	Vertices	
1	180	79.9991	0.00000	39.9995	0.360000	4
2	180	79.9991	0.00000	39.9995	0.360000	4
3	180	79.9991	0.00000	39.9995	0.360000	4
4	180	118.000	0.00000	59.0000	0.360000	4
5	180	118.000	0.00000	59.0000	0.360000	4
6	180	118.000	0.00000	59.0000	0.360000	4
7	180	118.000	0.00000	59.0000	0.360000	4
8	180	118.000	0.00000	59.0000	0.360000	4
9	180	118.000	0.00000	59.0000	0.360000	4

1) Result ID 649 check EXT\_angle\_180, cell Mentor2: 2 Edges

width 0.36  
radians 0  
degrees 180  
distance 79.9991  
Radius 39.9995

2 Edges. coordinates in cell Mentor2

```
{ 782.4201 -1264.0620} { 782.4211 -1264.0620}
{ 782.4199 -1184.4229} { 782.4209 -1184.4229}
```

# Sample Rules

```
LAYER MAP 351 DATATYPE == 11 1351
layer cad 1351

VARIABLE maximum_radius 501.
VARIABLE minimum_grid 0.001
VARIABLE minimum_curvature_width 0.26
VARIABLE maximum_curvature_width 0.50
VARIABLE circle_width 0.358 //inner w==0.358 and end width w==0.360 um

AND_cad_base = AND cad >1
AND_cad_base {COPY AND_cad_base}

AND_cad = SIZE (EXTENTS AND_cad_base ) BY minimum_grid*2 //recommnd to use 2 grids when operating skew edges.
AND_cad {COPY AND_cad}

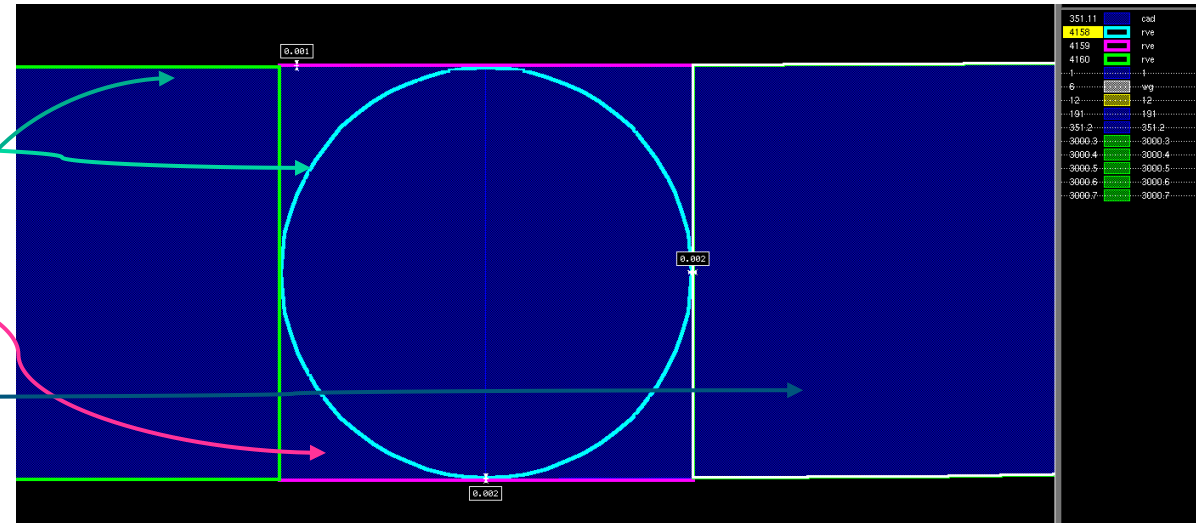
Adj_cad = cad NOT AND_cad
Adj_cad {COPY Adj_cad}

//Filter a half of cirlces at the end of polygones
//Value based on Area=Pi*(R^2) for a circle width(i.e. 0.358, 0.360) layer Adjust according to your technology dimensions.
Adj_cad_final = Adj_cad NOT (AREA Adj_cad < (3.1415926*(circle_width/2)*(circle_width/2)))
Adj_cad_final {COPY Adj_cad_final}
```

AND\_cad\_base

AND\_cad

Adj\_cad &  
Adj\_cad\_final



# Sample Rules

```
circle_props = DFM CIRCLE ANALYZE AND_cad_base
    circle_props {DFM COPY circle_props}

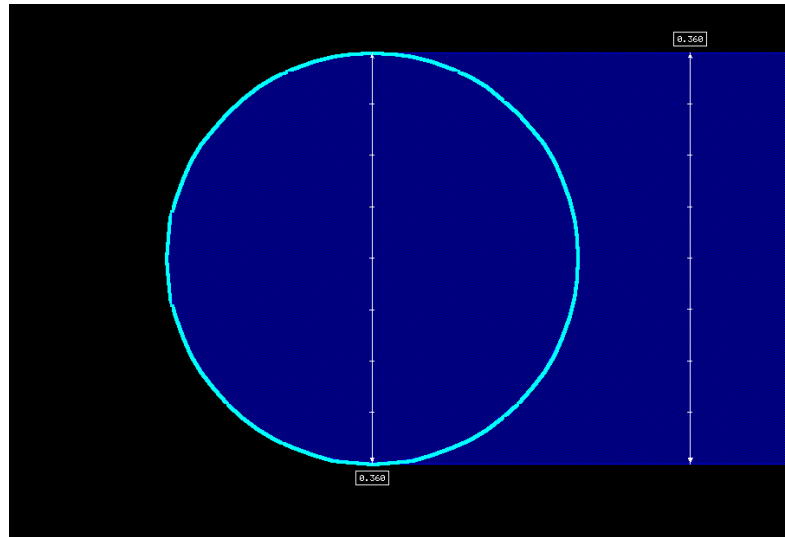
near_circle = DFM PROPERTY circle_props
    [inrad = PROPERTY(circle_props, INRADIUS)]
    [outrad = PROPERTY(circle_props, OUTRADIUS)]
    [annulus_in = PROPERTY(circle_props, ANNULUS_INRADIUS)]
    [ratio = PROPERTY_REF(inrad)/PROPERTY_REF(outrad)] > 0.90
    [width = (PROPERTY_REF(annulus_in)!=0) ? (PROPERTY_REF(inrad)+PROPERTY_REF(outrad))/2 - PROPERTY_REF(annulus_in): PROPERTY_REF(inrad)+PROPERTY_REF(outrad)] <= maximum_curvature_width
    [radius = (PROPERTY_REF(annulus_in)!=0) ? (PROPERTY_REF(width))+PROPERTY_REF(annulus_in): (PROPERTY_REF(inrad)+PROPERTY_REF(outrad))/2] <= maximum_curvature_width
near_circle { DFM COPY near_circle DFM RDB near_circle "dfm.rdb" ALL CELLS CHECKNAME "%_l_"}

```

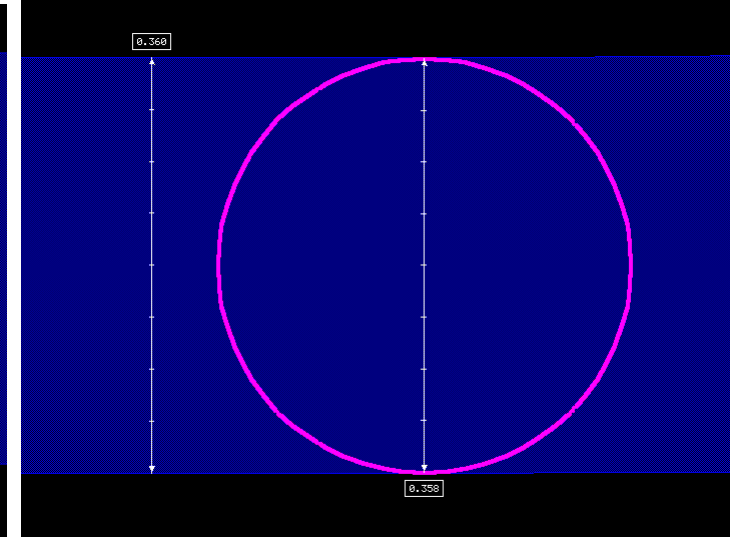
Check / Cell	Results
Check P_length_Adj_prop::<2>_1	419
Check near_circle	229
Check EXT_angle_180	69
Check out_corner	54
Check INT_angle	28
Check EXT_angle	0

	annulus_in	inrad	outrad	radius	ratio	width
1	0	0.179462	0.180427	0.179944	0.994648	0.359889
2	0	0.179462	0.180427	0.179944	0.994648	0.359889
3	0	0.179462	0.180427	0.179944	0.994648	0.359889
4	0	0.179462	0.180427	0.179944	0.994648	0.359889
5	0	0.179462	0.180427	0.179944	0.994648	0.359889
6	0	0.179462	0.180427	0.179944	0.994648	0.359889
7	0	0.179462	0.180427	0.179944	0.994648	0.359889
8	0	0.179462	0.180427	0.179944	0.994648	0.359889
9	0	0.179462	0.180427	0.179944	0.994648	0.359889
10	0	0.179462	0.180427	0.179944	0.994648	0.359889
11	0	0.179462	0.180427	0.179944	0.994648	0.359889
12	0	0.179462	0.180427	0.179944	0.994648	0.359889
13	0	0.179462	0.180427	0.179944	0.994648	0.359889
14	0	0.179462	0.180427	0.179944	0.994648	0.359889
15	0	0.178558	0.179605	0.179082	0.994170	0.358163



Circle Width ==0.360



Circle Width ==0.358

# Sample Rules

```

// Main codes start here
P_length = CONVEX EDGE Adj cad final WITH LENGTH <= maximum_curvature_width ANGLE1 <=95 ANGLE2 <=95
P_length {COPY P_length }

//Expand edge to compensate the a half of circle size + 0.001
P_length_ends = EXPAND EDGE P_length OUTSIDE BY (circle_width/2 + minimum_grid)
P_length_ends {COPY P_length_ends}

//Transfer property of P_length for single-edge error cluster
P_length_Adj = DFM SHIFT EDGE P_length OUTSIDE BY minimum_grid*2 SINGLE
P_length_Adj {DFM COPY P_length_Adj}

//P_length_Adj2 = DFM EXPAND EDGE P_length_Adj0 OUTSIDE BY minimum_grid*2
P_length_Adj2 = DFM EXPAND EDGE P_length OUTSIDE BY minimum_grid*2
P_length_Adj2 {DFM COPY P_length_Adj2}

P_length_Adj_prop = DFM PROPERTY P_length_Adj near_circle OVERLAP ABUT ALSO SINGULAR MULTI
[radius_minusgrid = PROPERTY(near_circle, radius) -0.001] > 0
[radius = PROPERTY(near_circle, radius)]

P_length_Adj_prop {DFM COPY P_length_Adj_prop DFM RDB P_length_Adj_prop dfm.rdb }

P_ends_exp = DFM EXPAND EDGE P_length_Adj_prop OUTSIDE BY "radius_minusgrid" // 0.179 -0.001 == 0.178
P_ends_exp {COPY P_ends_exp}

OR_P_ends_exp = OR P_ends_exp Adj cad final P_length_Adj2 P_length_ends // OR Boolean with four derived layers
OR_P_ends_exp {COPY OR_P_ends_exp}

//P_ends = CONVEX EDGE OR_P_ends_exp WITH LENGTH <= maximum_curvature_width ANGLE1 <=95 ANGLE2 <=95
P_ends = CONVEX EDGE OR_P_ends_exp ANGLE1 <=95 ANGLE2 <=95
P_ends {COPY P_ends}

```

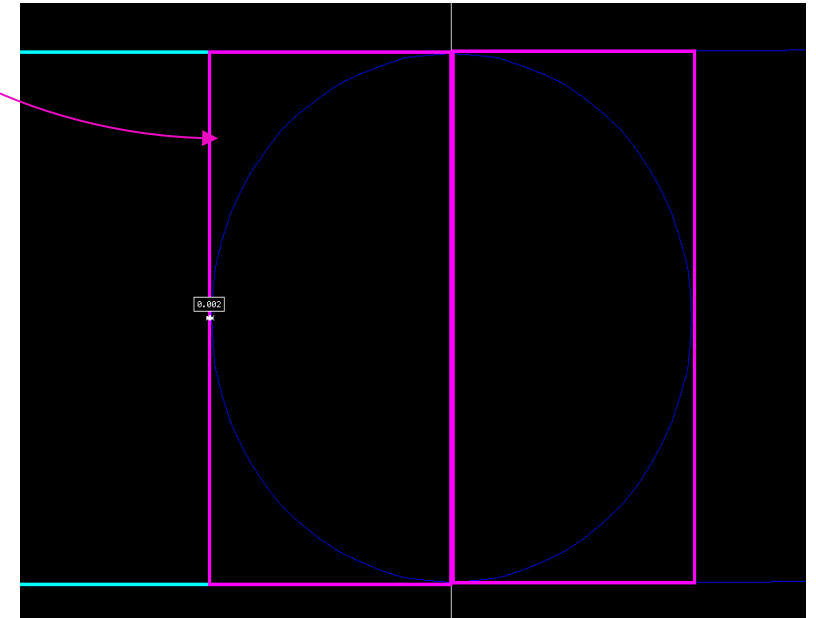
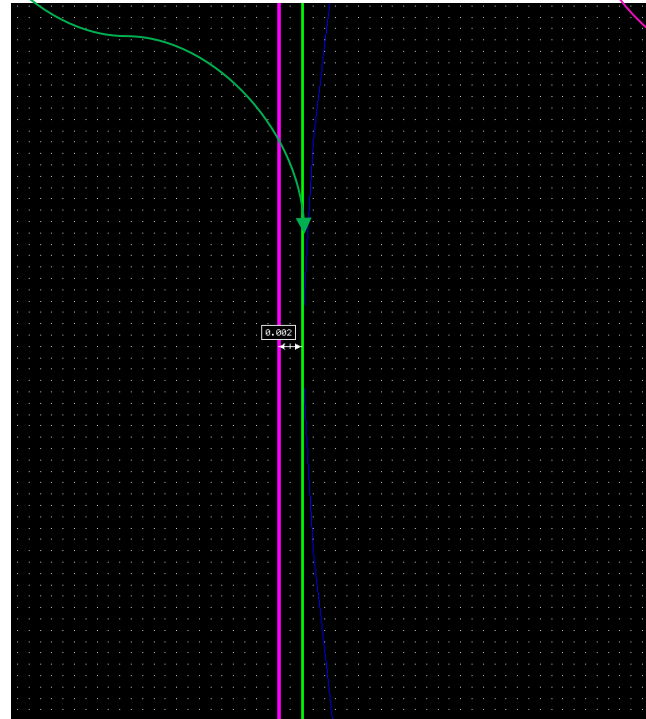
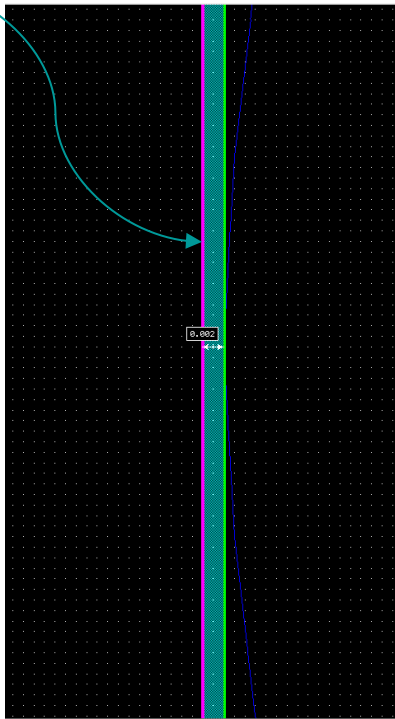
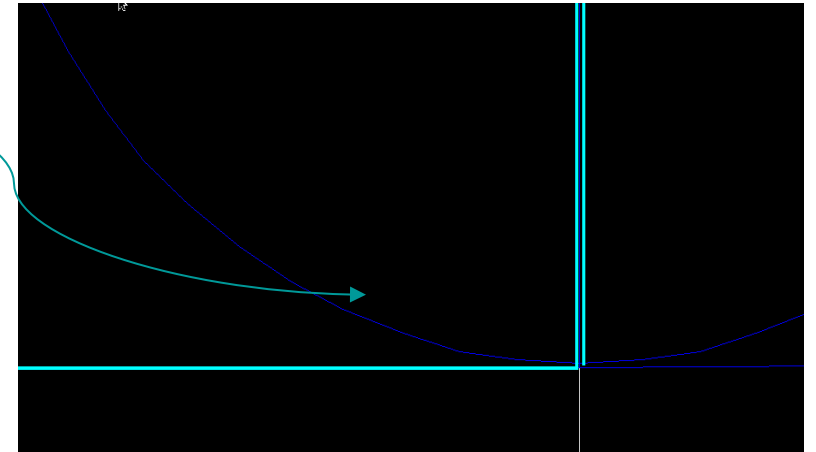
# Sample Rules

```
// Main codes start here
P_length = CONVEX EDGE Adj cad final WITH LENGTH <= maximum_curvature_width ANGLE1 <=95 ANGLE2 <=95
P_length {COPY P_length }

//Expand edge to compensate the a half of circle size + 0.001
P_length_ends = EXPAND EDGE P_length OUTSIDE BY (circle_width/2 + minimum_grid)
P_length_ends {COPY P_length_ends}

//Transfer property of P_length for single-edge error cluster
P_length Adj = DFM SHIFT_EDGE P_length OUTSIDE BY minimum_grid*2 SINGLE
P_length Adj {DFM COPY P_length Adj}

//P_length Adj2 = DFM EXPAND_EDGE P_length Adj0 OUTSIDE BY minimum_grid*2
P_length Adj2 = DFM EXPAND_EDGE P_length OUTSIDE BY minimum_grid*2
P_length Adj2 {DFM COPY P_length Adj2}
```



# Sample Rules

```

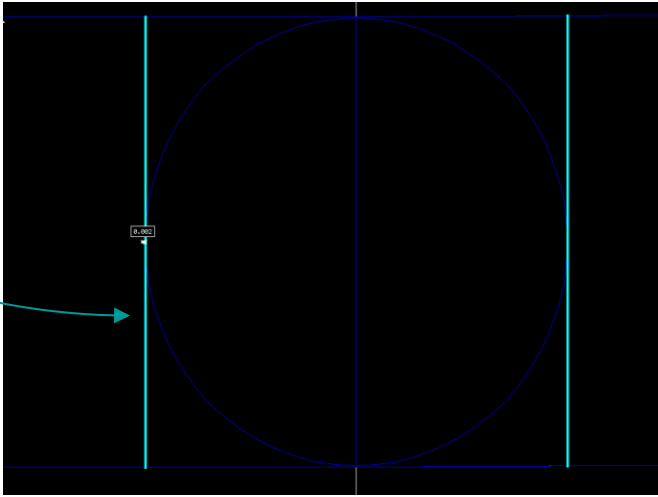
P_length_Adj_prop = DFM PROPERTY P_length_Adj near_circle OVERLAP ABUT ALSO SINGULAR MULTI
[radius_minusgrid = PROPERTY(near_circle, radius) -0.001] > 0
[radius = PROPERTY(near_circle, radius)]

P_length_Adj_prop {DFM COPY P_length_Adj_prop DFM RDB P_length_Adj_prop dfm.rdb }

P_ends_exp = DFM EXPAND EDGE P_length_Adj_prop OUTSIDE BY "radius_minusgrid" // 0.179 -0.001 == 0.178
P_ends_exp {COPY P_ends_exp}

OR_P_ends_exp = OR P_ends_exp Adj_cad_final P_length_Adj2 P_length_ends // OR Boolean with four derived layers
OR_P_ends_exp {COPY OR_P_ends_exp}

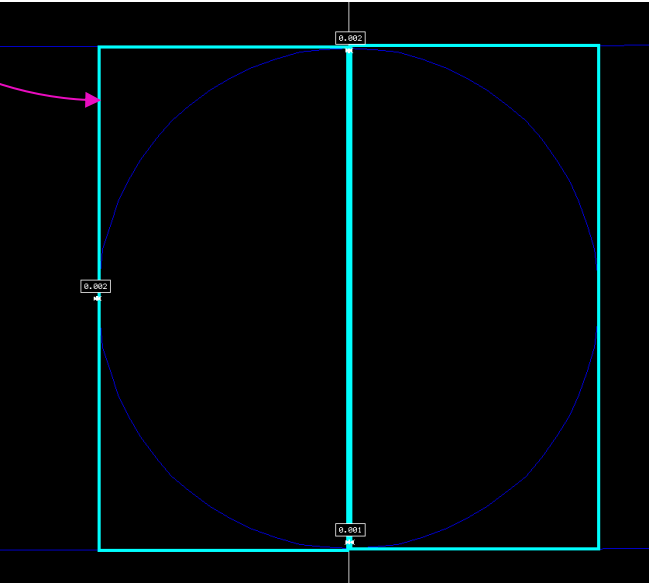
//P_ends = CONVEX EDGE OR_P_ends_exp WITH LENGTH <= maximum_curvature_width ANGLE1 <=95 ANGLE2 <=95
P_ends = CONVEX EDGE OR_P_ends_exp ANGLE1 <=95 ANGLE2 <=95
P_ends {COPY P_ends}
    
```



dfm.rdb x drc.db

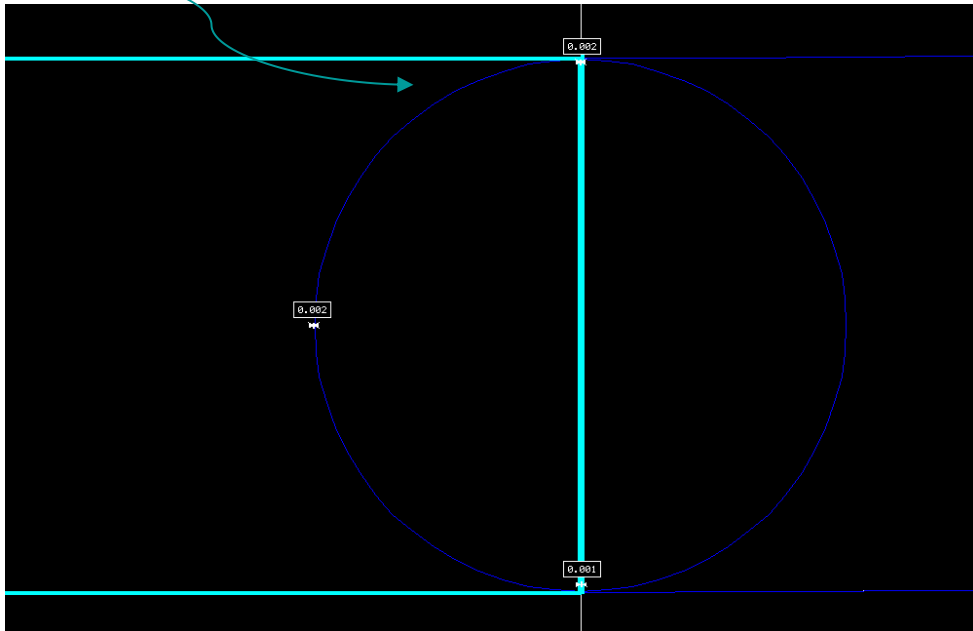
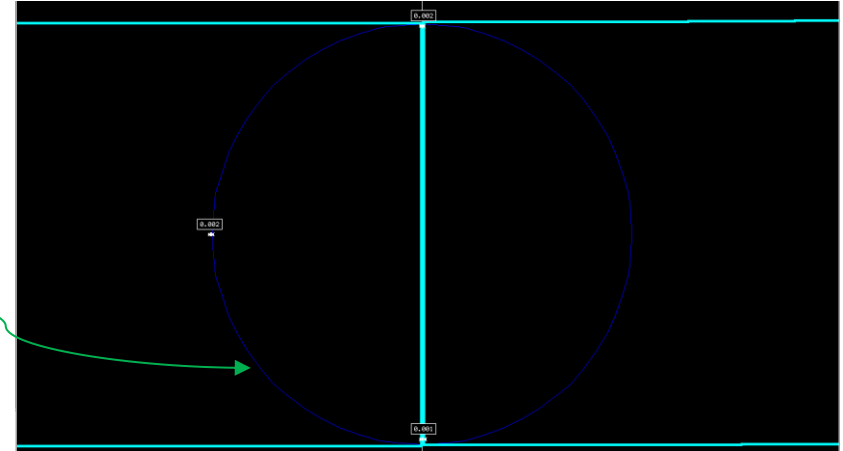
Filter: Show All Mentor2, 799 Results (in 5 of 6 Checks)

Check / Cell	Results	radius	radius_minusgrid
Check P_length_Adj_prop::<2>_1	419	1 0.179082	0.178082
Check near_circle	229	2 0.179082	0.178082
Check EXT_angle_180	69	3 0.179082	0.178082
Check out_corner	54	4 0.179082	0.178082
Check INT_angle	28	5 0.179082	0.178082
Check EXT_angle	0	6 0.179082	0.178082
		7 0.179082	0.178082

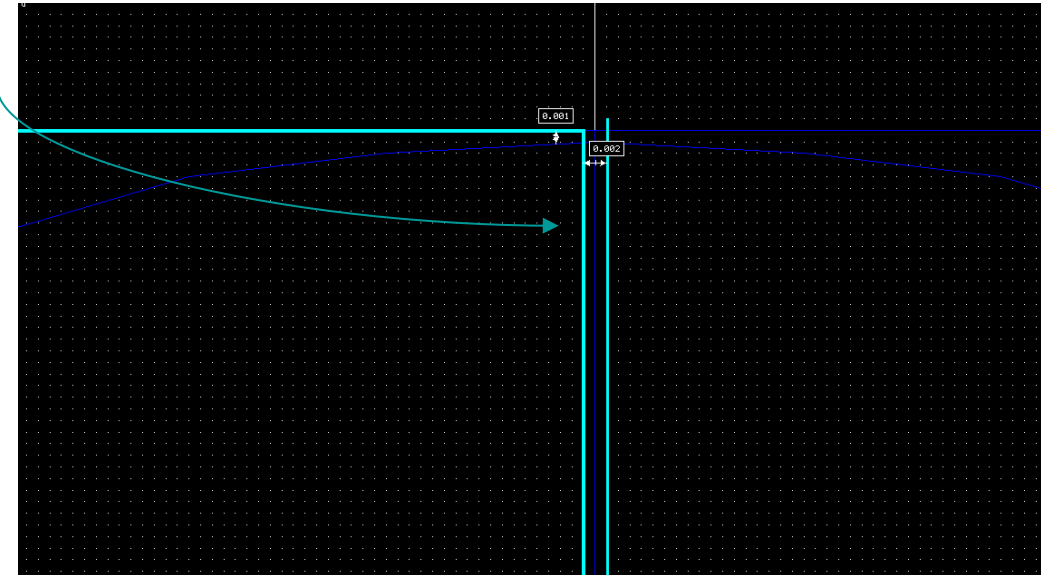


# Sample Rules

```
P_length Adj_prop = DFM PROPERTY P_length Adj near_circle OVERLAP ABUT ALSO SINGULAR MULTI  
[radius_minusgrid = PROPERTY(near_circle, radius) -0.001] > 0  
[radius = PROPERTY(near_circle, radius)]  
  
P_length Adj_prop {DFM COPY P_length Adj_prop DFM RDB P_length Adj_prop dfm.rdb }  
  
P_ends_exp = DFM EXPAND EDGE P_length Adj_prop OUTSIDE BY "radius_minusgrid" // 0.179 -0.001 == 0.178  
P_ends_exp {COPY P_ends_exp}  
  
OR_P_ends_exp = OR P_ends_exp Adj_cad_final P_length Adj2 P_length_ends // OR Boolean with four derived layers  
OR_P_ends_exp {COPY OR_P_ends_exp}  
  
//P_ends = CONVEX EDGE OR_P_ends_exp WITH LENGTH <= maximum_curvature_width ANGLE1 <=95 ANGLE2 <=95  
P_ends = CONVEX EDGE OR_P_ends_exp ANGLE1 <=95 ANGLE2 <=95  
P_ends {COPY P_ends}
```



Zoom-In



# 3. Suggestion

- Same Diameter of Circle to write a simple rule

For example, Inner Circle width == 0.360 um

- Benefit:
  - Users can assume they can make code shorter and reduce runtime
  - Accurate property value can be obtained
- No separate license is required for DFM CIRCLE ANALYZE