```
icc2 shell> man set floorplan enclosure rules
 2
 3
     2. Synopsys Commands
                                                                   Command Reference
 4
                              set floorplan enclosure rules
 5
 6
    NAME
 7
            set floorplan enclosure rules
8
                   Defines a floorplan enclosure rule.
9
10
    SYNTAX
11
            status set floorplan enclosure rules
12
                   -from object types from type list
                   -to object types to_type_list
13
14
                   -to lib cells lib cells
15
                   -sides side list
16
                   -from corner corner type
17
                   -to corner corner type
18
                   -name rule name
19
                   [-must enclose]
20
                   [-follow rotations]
21
                   [-ignore rotate90]
22
                   [-layers layer list]
23
                   [-to layers layer list]
24
                   [-forbidden list distance list]
25
                   [-forbidden ranges {{low high} {low1 high1} ... }]
26
                   [-max distance]
27
                   [-min distance]
28
                   [-offset distance]
29
                   [-step distance]
30
                   [-valid list distance list]
31
                   [-valid ranges {{low high} {low1 high1} ... }]
32
                   [-offset ranges {{low high} {low1 high1} ... }]
33
        Data Types
34
35
            from type list list
36
            to type list
                            list
37
            lib cells
                            collection
38
            side list
                            list
39
            corner_type
                            string
40
            rule name
                            string
41
            layer list
                           list
42
            distance list list
43
            low
                           float
44
            high
                            float
45
            low1
                            float
46
            high1
                            float
47
            distance
                            float
48
49
    ARGUMENTS
50
            -from_object_types from_type_list
                   Specifies the \overline{\text{list}} of "from" object types \mathbf{for} the enclosure
51
52
                   floorplan rule. These type of objects enclose the object types
53
                   specified with -to object types or library cells specified with
54
                   -to lib cells. Valid values for this option are block boundary,
55
                                placement blockage, routing blockage, unplace-
                   core area,
56
                   able_area, std_cell_area, va_boundary, shape and cover_bump.
57
                   This is a mandatory option.
58
59
            -to object types to type list
                   Specifies the list of "to" object types for the enclosure floor-
60
61
                   plan rule. These type of objects are enclosed by other objects
62
                   specified with -from object types. Valid values for this option
63
                   are core area, hard macro, placement_blockage, routing_blockage,
64
                   soft macro, unplaceable area, std cell area, va boundary, shape
65
                   and cover bump. This option is mutually exclusive
66
                   -to lib cells and you must specify one or the other.
67
68
            -to lib cells lib cells
                   Specifies the collection of library cells for the enclosure
69
```

floorplan rule. These library cells are being enclosed by other 71 objects specified in -from object types. This option is mutually 72 exclusive with -to object types and either one of them must be 73 specified. 74 75 -sides side list 76 Specifies the sides or directions from which the "from" object 77 encloses the "to" object. Spacing is checked between the 78 objects. Valid values are all, bottom, horizontal, left, right, 79 top, vertical and manhattan. The horizontal argument includes 80 both left and right. Similarly, the vertical argument includes both bottom and top. This option is mutually exclusive with the 81 82 -from corner and -to corner option pair. Either this option or 83 -from corner and -to corner pair must be specified. 84 85 -from corner corner type 86 Specifies the corner from which the "from" object encloses the 87 "to" object. Spacing is checked between the objects. Valid val-88 ues are all, bottom left, bottom right, top left and top right. 89 This option can be used when spacing must be checked from a cor-90 ner instead of an edge. This option must be used together with 91 -to corner and is mutually exclusive with -sides. 92 93 -to corner corner type Specifies the corner from which the "to" object is being enclosed by "from" object. Spacing is checked between the objects. Valid values are all, bottom_left, bottom_right, 94 95 96 97 top_left and top_right. This option can be used when spacing 98 must be checked from a corner instead of an edge. This option 99 must be used together with -from corner and is mutually exclu-100 sive with -sides. 101 102 -name rule name 103 104 mandatory option. 105 106 -must enclose 107 108 109 110 -follow rotations 111

Specifies the name of the enclosure floorplan rule. This is a

Specifies that the "from" object must completely enclose the "to" object from all sides. This is an optional option.

Specifies whether the sides specified by the -sides option should follow the rotations of library cells, that is, if the meaning of horizontal or vertical should change when the library cell has a 90-degree rotation. This option must be used together -to lib cells or -to object types hard macro or -to object types soft macro. This is an optional option.

-ignore rotate90

112

113

114

115

116 117 118

119

120

121 122

123 124

125

126

127

128 129 130

131

132

133

134

135 136

137

138

Specifies whether this rule can be ignored for library cells with 90-degree rotations. This option must be used together with -to lib cells **or** -to object types hard macro **or** -to object types soft macro. This is an optional option.

-layers layer list

Specifies the routing layers to be considered for "from" routing blockage or shape object type. This option must be used -from object types routing blockage with -from object types shape. This is an optional option.

-to_layers layer_list

Specifies the routing layers to be considered for "to" routing blockage or shape object type. This option must be used along with -to object types routing blockage or -to object types shape. This is an optional option.

-forbidden list distance list

Specifies a list of distances by which the "from" object cannot enclose the "to" object. This option is mutually exclusive with

-valid list. Values in the distance list cannot be negative. 139 140 This is an optional option. 141 142 -forbidden ranges {{low high} {low1 high1} ... } 143 Specifies a list of distance ranges between which the "from" object cannot enclose the "to" object. The enclosing distance 144 145 must not lie within any of low and high in the specified list of 146 ranges. This option is mutually exclusive with -valid ranges. 147 Values cannot be negative. This is an optional option. 148 -max distance Specifies the maximum distance by which the "from" object can 150 151 enclose the "to" object. The distance cannot be greater than this value. The distance cannot be negative. If -min is also 152 153 specified, this value must be greater than the min value. This 154 is an optional option. 155 156 -min distance 157 Specifies the minimum distance by which the "from" object can 158 enclose the "to" object. The distance cannot be less than this 159 value. The value specified cannot be negative. If -max is also 160 specified then this value must be less than the max value. This 161 is an optional option. 162 163 -offset distance 164 Specifies a parameter in distance calculation between the "from" 165 and "to" objects. This option must be used together with -step. 166 This implies that the distance has to be an integer multiple of 167 the -step value plus the -offset value. The value specified can-168 not be negative. This option is mutually exclusive with -off-169 set ranges. This is an optional option. 170 171 -step distance 172 Specifies a parameter in distance calculation between the "from" 173 and "to" objects. This option must be used together with -offset 174 or -offset ranges. This implies that the distance has to be an 175 integer multiple of the -step value plus the -offset value or 176 distance has to be in range of an integer multiple of the -step 177 value plus the -offset_ranges value. The value specified must be 178 greater than zero. This is an optional option. 179 -valid list distance list 180 181 Specifies a list of distances by which the "from" object can 182 enclose the "to" object. This option is mutually exclusive with 183 -forbidden list. Values specified cannot be negative. This is an 184 optional option. 185 186 -valid ranges {{low high} {low1 high1} ... } Specifies a list of distance ranges between which the "from" 187 object can enclose the "to" object. The distance must lie ${\bf within}$ 188 189 any of low and high in the specified list of ranges. This option 190 is mutually exclusive with -forbidden ranges. Values specified 191 cannot be negative. This is an optional option. 192 -offset ranges {{low high} {low1 high1} ... } 194 Specifies a list of distance ranges. This implies that the dis-195 tance has to be in range of an integral multiple of step value plus offset ranges value. Values specified can't be negative. 196 This option must be used along with -step. This is an optional 197 198 option. 199 200 DESCRIPTION 201 The set floorplan enclosure rules command defines a named enclosure 202 floorplan rule in the current design. The defined rule is persistent. 203 If another floorplan rule by the same name exists then the command 204 errors out. 205 206 There is a difference between the object types core area and std cell area. The core area object type is the core boundary region 207

without cutting out any blockages and is typically applicable for the top level. The std cell area object type is the core boundary region after cutting out all blockages and is typically applicable for the block level.

If an enclosure rule is defined for a library cell and another enclosure rule is defined for a hard macro, the enclosure rule defined for the library cell takes precedence over the enclosure rule defined for hard macro when checks are done for that library cell.

If the measured value falls **inside** valid range **or** is a member of the valid list then there is no violation given by check floorplan rules regardless of other constraints like min, max, and so on. If this measured value is outside valid range or list then a violation is reported if other constraints are specified and they are not met or if no other constraints are specified.

225 EXAMPLES

208 209

210

211

212 213

214

215

216

217 218

219

220

221

222

223

224

226 227

228

229

230 231

232 233

234

235

236

237

238

240

241 242

243

244

245

246

247

248

249

250 251

252

253

254

icc2 shell>

The following example creates an enclosure rule named e1. This rule checks the enclosing of hard macros and core area by the block boundary. The check is done for horizontal sides, both left and right, and only at the bottom in vertical side. Must be greater than 2 and less than 100, but cannot be in the range between 5 and 15. Also, the distance must be an **integer** multiple of 2 plus 10.

```
prompt> set floorplan enclosure rules -name e1 \
                  -from object types block boundary \
                  -to_object_types {hard_macro core_area} -sides {bottom horizontal} \
                  -forbidden ranges {{5 15}} -valid list {2 20 30} -max 100 -min 2 \
                  -offset 10 -step 2
239 SEE ALSO
            remove floorplan rules (2)
             report floorplan rules (2)
```

set floorplan area rules (2) set floorplan extension rules (2) set floorplan exception rules (2) set floorplan forbidden rules (2) set_floorplan_halo_rules(2) set floorplan length rules (2) set floorplan spacing rules (2) set floorplan_width_rules(2)

> Version S-2021.06-SP5 Copyright (c) 2022 Synopsys, Inc. All rights reserved.