```
icc2_shell> man set_floorplan_enclosure_rules

2.  Synopsys Commands                                    Command Reference
                        set_floorplan_enclosure_rules

NAME
       set_floorplan_enclosure_rules
              Defines a floorplan enclosure rule.

SYNTAX
       status set_floorplan_enclosure_rules
              -from_object_types from_type_list
              -to_object_types to_type_list
              -to_lib_cells lib_cells
              -sides side_list
              -from_corner corner_type
              -to_corner corner_type
              -name rule_name
              [-must_enclose]
              [-follow_rotations]
              [-ignore_rotate90]
              [-layers layer_list]
              [-to_layers layer_list]
              [-forbidden_list distance_list]
              [-forbidden_ranges {{low high} {low1 high1} ... }]
              [-max distance]
              [-min distance]
              [-offset distance]
              [-step distance]
              [-valid_list distance_list]
              [-valid_ranges {{low high} {low1 high1} ... }]
              [-offset_ranges {{low high} {low1 high1} ... }]

   Data Types
       from_type_list  list
       to_type_list    list
       lib_cells       collection
       side_list       list
       corner_type     string
       rule_name       string
       layer_list      list
       distance_list   list
       low             float
       high            float
       low1            float
       high1           float
       distance        float

ARGUMENTS
       -from_object_types from_type_list
              Specifies the list of "from" object types for the enclosure
              floorplan rule. These type of objects enclose the object types
              specified with -to_object_types or library cells specified with
              -to_lib_cells. Valid values for this option are block_boundary,
              core_area, placement_blockage, routing_blockage, unplace-
              able_area, std_cell_area, va_boundary, shape and cover_bump.
              This is a mandatory option.

       -to_object_types to_type_list
              Specifies the list of "to" object types for the enclosure floor-
              plan rule. These type of objects are enclosed by other objects
              specified with -from_object_types. Valid values for this option
              are core_area, hard_macro, placement_blockage, routing_blockage,
              soft_macro, unplaceable_area, std_cell_area, va_boundary, shape
              and cover_bump. This option is mutually exclusive with
              -to_lib_cells and you must specify one or the other.

       -to_lib_cells lib_cells
              Specifies the collection of library cells for the enclosure
```

floorplan rule. These library cells are being enclosed by other objects specified in -from_object_types. This option is mutually exclusive with -to_object_types and either one of them must be specified.

-sides side_list
Specifies the sides or directions from which the "from" object encloses the "to" object. Spacing is checked between the objects. Valid values are all, bottom, horizontal, left, right, top, vertical and manhattan. The horizontal argument includes both left and right. Similarly, the vertical argument includes both bottom and top. This option is mutually exclusive with the -from_corner and -to_corner option pair. Either this option or -from_corner and -to_corner pair must be specified.

-from_corner corner_type
Specifies the corner from which the "from" object encloses the "to" object. Spacing is checked between the objects. Valid values are all, bottom_left, bottom_right, top_left and top_right. This option can be used when spacing must be checked from a corner instead of an edge. This option must be used together with -to_corner and is mutually exclusive with -sides.

-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, top_left and top_right. This option can be used when spacing must be checked from a corner instead of an edge. This option must be used together with -from_corner and is mutually exclusive with -sides.

-name rule_name
Specifies the name of the enclosure floorplan rule. This is a mandatory option.

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

-follow_rotations
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 with -to_lib_cells or -to_object_types hard_macro or -to_object_types soft_macro. This is an optional option.

-ignore_rotate90
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 along with -from_object_types routing_blockage or -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. This is an optional option.

-forbidden_ranges {{low high} {low1 high1} ... }
Specifies a list of distance ranges between which the "from" object cannot enclose the "to" object. The enclosing distance must **not** lie **within** any of low **and** high in the specified list of ranges. This option is mutually exclusive **with** -valid_ranges. Values cannot be negative. This is an optional option.

-max distance
Specifies the maximum distance by which the "from" object can enclose the "to" object. The distance cannot be greater than **this** value. The distance cannot be negative. If -min is also specified, **this** value must be greater than the min value. This is an optional option.

-min distance
Specifies the minimum distance by which the "from" object can enclose the "to" object. The distance cannot be less than **this** value. The value specified cannot be negative. If -max is also specified then **this** value must be less than the max value. This is an optional option.

-offset distance
Specifies a **parameter** in distance calculation between the "from" **and** "to" objects. This option must be used together **with** -step. This **implies** that the distance has to be an **integer** multiple of the -step value plus the -offset value. The value specified can-**not** be negative. This option is mutually exclusive **with** -off-set_ranges. This is an optional option.

-step distance
Specifies a **parameter** in distance calculation between the "from" **and** "to" objects. This option must be used together **with** -offset **or** -offset_ranges. This **implies** that the distance has to be an **integer** multiple of the -step value plus the -offset value **or** distance has to be in range of an **integer** multiple of the -step value plus the -offset_ranges value. The value specified must be greater than zero. This is an optional option.

-valid_list distance_list
Specifies a list of distances by which the "from" object can enclose the "to" object. This option is mutually exclusive **with** -forbidden_list. Values specified cannot be negative. This is an optional option.

-valid_ranges {{low high} {low1 high1} ... }
Specifies a list of distance ranges between which the "from" object can enclose the "to" object. The distance must lie **within** any of low **and** high in the specified list of ranges. This option is mutually exclusive **with** -forbidden_ranges. Values specified cannot be negative. This is an optional option.

-offset_ranges {{low high} {low1 high1} ... }
Specifies a list of distance ranges. This **implies** that the dis-tance has to be in range of an integral multiple of step value plus offset_ranges value. Values specified can't be negative. This option must be used along **with** -step. This is an optional option.

DESCRIPTION

The set_floorplan_enclosure_rules command defines a named enclosure floorplan rule in the current **design**. The defined rule is persistent. If another floorplan rule by the same name exists then the command errors out.

There is a difference between the object types core_area **and** std_cell_area. The core_area object **type** is the core boundary region

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.

EXAMPLES

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
```

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)

icc2_shell>