```
icc2_shell> man set_floorplan_location_rules

2.  Synopsys Commands                                    Command Reference
                          set_floorplan_location_rules

NAME
       set_floorplan_location_rules
              Defines a location floorplan rule in the design.

SYNTAX
       status set_floorplan_location_rules
              -from_object_types type_list
              -to_object_types type_list
              -from_lib_cells lib_cells
              -to_lib_cells lib_cells
              -from_type location_type
              -to_type location_type
              [-follow_rotations]
              [-ignore_rotate90]
              -direction direction
              [-from_layers layer_list]
              [-to_layers layer_list]
              -name rule_name
              [-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
       type_list              list
       lib_cells              collection
       location_type          string
       direction              string
       layer_list             list
       rule_name              string
       distance_list          list
       low                    float
       high                   float
       low1                   float
       high1                  float
       distance               string

ARGUMENTS
       -from_object_types type_list
              Specifies the list of "from" object types for the location
              floorplan rule. Location from these type of objects for a "from"
              location type is checked with other objects specified with
              -to_object_types or library cells specified with -to_lib_cells.
              Valid values for this option are block_boundary, core_area,
              hard_macro, placement_blockage, routing_blockage, shape,
              soft_macro and std_cell_area. This option is mutually exclusive
              with -from_lib_cells and you must specify one or the other.

       -to_object_types type_list
              Specifies the list of "to" object types for the location floor-
              plan rule. Location to these type of objects for a "to" location
              type is checked with other objects specified with
              -from_object_types or library cells specified with
              -from_lib_cells. Valid values for this option are block_bound-
              ary, core_area, hard_macro, placement_blockage, routing_block-
              age, shape, soft_macro and std_cell_area. This option is mutu-
              ally exclusive with -to_lib_cells and you must specify one or
              the other.
```

-from_lib_cells lib_cells
Specifies the collection of "from" lib cells for the location floorplan rule. Location from these lib cells for a "from" location type is checked with other objects specified with -to_object_types or library cells specified with -to_lib_cells. This option is mutually exclusive with -from_object_types and you must specify one or the other.

-to_lib_cells lib_cells
Specifies the collection of "to" lib cells for the location floorplan rule. Location to these lib cells for a "to" location type is checked with other objects specified with -from_object_types or library cells specified with -from_lib_cells. This option is mutually exclusive with -to_object_types and you must specify one or the other.

-from_type location_type
Specifies the "from" location type of the location check. Valid values are all_corners, bbox_all, bbox_bottom_left, bbox_top_right, bottom_left , bottom_right, top_left and top_right. This is a mandatory option.

-to_type location_type
Specifies the "to" location type of the location check. Valid values are all_corners, bbox_all, bbox_bottom_left, bbox_top_right, bottom_left , bottom_right, top_left and top_right. This is a mandatory option.

-follow_rotations
Specifies whether mentioned direction should follow the rotations of library cells, that is, if meaning of horizontal or vertical should change when library cell has a 90-degree rotations. This option must be used together with -from_lib_cells or -to_lib_cells or -to_object_types hard_macro or -to_object_types soft_macro or -from_object_types hard_macro or -from_object_types soft_macro. This is an optional option.

-ignore_rotate90
Specifies whether this rule can be ignored for library cells with a 90-degree rotation. This option must be used together with -from_lib_cells or -to_lib_cells or -to_object_types hard_macro or -to_object_types soft_macro or -from_object_types hard_macro or -from_object_types soft_macro. This is an optional option.

-direction direction
Specifies the side or direction in which location of object or lib cells needs to be checked. Valid values are horizontal and vertical. The horizontal stands for both left and right. Similarly the vertical stands for both bottom and top. This is a mandatory option.

-from_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 option. 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 option. This is an optional option.

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

-forbidden_list distance_list

Specifies a list of distance that should **not** be location of object **or** lib **cell for** the location **type**. This option is mutually exclusive **with** -valid_list. Values specified can't be negative. This is an optional option.

-forbidden_ranges **{{**low high**} {**low1 high1**} ... }**
Specifies a list of distance ranges that should **not** be location of object **or** lib **cell for** the location **type**. The distance must **not** lie **within** any of low **and** high in the specified list of ranges. This option is mutually exclusive **with** -valid_ranges. option. Values specified can't be negative. This is an optional option.

-max distance
Specifies the maximum location of object **or** lib **cell**. The distance can't be greater than **this** value. Value specified can't be negative. If -min is also specified then **this** value must be greater than the min value. This is an optional option.

-min distance
Specifies the minimum location of object **or** lib **cell**. The distance can't be lesser than **this** value. Value specified can't be negative. If -max is also specified then **this** value must be lesser than the max value. This is an optional option.

-offset distance
Specifies a **parameter** in location calculation of object **or** lib **cell**. This option must be used along **with** -step. This **implies** that the location has to be an integral multiple of step value plus offset value. Value specified can't be negative. This option is mutually exclusive **with** -offset_ranges. This is an optional option.

-step distance
Specifies a **parameter** in location calculation of object **or** lib **cell**. This option must be used along **with** -offset **or** -offset_ranges. This **implies** that the location has to be an integral multiple of step value plus offset value **or** location has to be in range of an integral multiple of step value plus offset_ranges value. Value specified must be greater than zero. This is an optional option.

-valid_list distance_list
Specifies a list of distance that should be location of object **or** lib **cell for** the location **type**. This option is mutually exclusive **with** -forbidden_list. Values specified can't be negative. This is an optional option.

-valid_ranges **{{**low high**} {**low1 high1**} ... }**
Specifies a list of distance ranges that should be location of object **or** lib **cell for** the location **type**. 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 can't be negative. This is an optional option.

-offset_ranges **{{**low high**} {**low1 high1**} ... }**
Specifies a list of distance ranges. This **implies** that the location 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_location_rules command defines a named location 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 **type** core_area **and**

std_cell_area. The core_area object type means core boundary region without cutting out any blockages and is typically applicable for top level whereas std_cell_area object type means core boundary region after cutting out all blockages and is typically applicable for block level.

If a location rule is defined for a lib cell and another location rule is defined for a hard macro then the location rule defined for the lib cell takes precedence over the location rule defined for hard macro when checks are done for that lib 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 a location rule by name l1 to check the locations of a soft macro and a placement blockage such that top-right of the soft macro is vertically (both the bottom and the top) within a maximum distance of 2300 of the bottom-right of the placement blockage.

```
prompt> set_floorplan_location_rules -name l1 -from_object_types soft_macro \
   -from_type top_right -to_object_types placement_blockage -to_type
   bottom_right \
   -direction vertical -max 2300
```

SEE ALSO

set_floorplan_area_rules(2)
set_floorplan_enclosure_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)
remove_floorplan_rules(2)
report_floorplan_rules(2)

```
                        Version S-2021.06-SP5
            Copyright (c) 2022 Synopsys, Inc. All rights reserved.
icc2_shell>
```