#ICC2 : Floorplan Automated Script

Tool : Synopsys IC Compiler II (ICC2)

Stage : Floorplan

Date : 24-01-2026

Open import design

```
open_lib ORCA_TOP.nlib
copy_block -from_block import_design -to_block floorplan
open_block floorplan
```

#Core & Die Area Creation

#Defines utilization, offset, and chip shape before initializing floorplan.

```
# Core and die area setup
set utilization 0.75
set offset 5
set shape "L"
if {$shape == "L"} {
  set side_ratio {1 1 1 1}
}
```

#Initialize Floorplan

#Creates the core and die based on given parameters.

```
initialize_floorplan \
  -core_utilization $utilization \
  -core_offset $offset \
  -shape $shape \
  -use_site_row
```

#Clean Old Constraints (Optional)

#Removes any previous blockages and pin guides.

```
if 0 {
  remove_placement_blockages *
  remove_pin_guides *
}
```

#Port Placement

#Loads external script for port placement.

```
proc cord {coord} {
   set llx [lindex $coord 0 0]
   set lly [lindex $coord 0 1]
   set urx [lindex $coord 1 0]
   set ury [lindex $coord 1 1]
   set bbox [list [list $llx $lly] [list $urx $ury]]
   create_pin_guide -boundary $bbox -layers {M3 M5} -pin_spacing 5 [all_inputs]
   place_pin -ports [all_inputs]
   return
}
proc cord1 {coord} {
   set llx [lindex $coord 0 0]
   set lly [lindex $coord 0 1]
   set urx [lindex $coord 1 0]
   set ury [lindex $coord 1 1]
   set bbox1 [list [list $llx $lly] [list $urx $ury]]
   set all_output [get_ports [all_outputs]]
   set out_list [get_object_name $all_output]
   set group1 [lrange $out_list 0 61]
   create_pin_guide -boundary $bbox1 -layers {M5} -pin_spacing 1 $group1
   place_pin -ports $group1
   return
}
proc cord2 {coord} {
   set llx [lindex $coord 0 0]
   set lly [lindex $coord 0 1]
   set urx [lindex $coord 1 0]
   set ury [lindex $coord 1 1]
   set bbox2 [list [list $llx $lly] [list $urx $ury]]
   set all_output [get_ports [all_outputs]]
   set out_list [get_object_name $all_output]
```

```
  set group2 [lrange $out_list 61 end]
  create_pin_guide -boundary $bbox2 -layers {M5} -pin_spacing 1 $group2
  place_pin -ports $group2
  return
}
```

#Coordinate Definitions

```
#Calling the proc
cord  {{0.0000 257.4720} {5.0000 429.4070}}
cord1 {{447.1680 529.5350} {452.1680 710.5840}}
cord2 {{889.3360 186.1770} {894.3360 326.0240}}
```

#Remove Voltage Areas (Optional)

#Deletes existing voltage areas if required.

```
if 0 {
  remove_voltage_area *
}
```

if 0 {

Voltage Area Creation

This section defines a procedure to calculate voltage area bounding box coordinates and create a voltage region.

Purpose:

- Adds offset to given coordinates.
- Creates a voltage area for a specific power domain.
- Adds guard band margins.

}

#Script:

```
proc va_cord {cord} {
  set llx [expr [lindex $cord 0 0] + 5.016]
  set lly [expr [lindex $cord 0 1] + 5.016]
  set urx [expr [lindex $cord 1 0] + 5.016]
  set ury [expr [lindex $cord 1 1] + 5.016 + 1.672]
  set va  [list [list $llx $lly] [list $urx $ury]]
  puts "voltage_bbox_area = $va"
```

```
  create_voltage_area -power_domains PD_RISC_CORE -region $va -guard_band {{5.016 5.016}}
  return
}
```

#Voltage Area Coordinates

#Defines the region for the voltage area.

#Calling proc

```
va_cord {{5.0000 5.0000} {402.3280 170.5280}}
```

#Macro Placement

#Enable Macro-Only Placement

#Only macros are placed during this stage, and standard cells are ignored.

```
set_app_options -name plan.macro.macro_place_only -value true
```

#Macros are grouped and placed according to design hierarchy.

```
set_app_options -name plan.macro.grouping_by_hierarchy -value true
```

#Minimum vertical spacing between macros is defined to allow routing and power straps.

```
set_app_options -name plan.macro.spacing_rule_heights -value {15um 15um}
```

#Minimum horizontal spacing between macros is defined.

```
set_app_options -name plan.macro.spacing_rule_widths -value {15um 15um}
```

if 0 {

optional app options

Macro placement can consider timing and hierarchy when enabled.

```
# plan.place.hierarchy_by_name
# plan.place.timing_driven_mode
```

}

#Macro placement is performed based on the floorplan.

```
create_placement -floorplan
```

#Sanity check: verify macros exist

```
if {[sizeof_collection [get_flat_cells -filter "is_hard_macro"]] == 0} {
  puts "WARNING: No hard macros found in the design!"
} else {
  puts "INFO: Hard macros detected: \
    [sizeof_collection [get_flat_cells -filter "is_hard_macro"]]"
}
```

#Keepout Margin Creation

#Adds a margin around hard macros to avoid standard cells being placed too close.

```
if {$macro_cnt > 0} {
   create_keepout_margin -outer {1.5 1.5 1.5 1.5} $macro_cells
}
set_app_option -name place.floorplane.sliver.size -value 4um
derive_placement_blockages -f
redirect ./scripts/par_pl.txt {get_attr [get_placement_blockages] bbox}
remove_placement_blockages *
```

#Partial Blockage Creation from File

#Reads the saved file and creates partial placement blockages.

```
set fh [open "/home/rajshekhars/raj_scripts/scripts/par_pl.txt" r]
set fb [read $fh]
foreach pl $fb {
   create_placement_blockage -type partial -boundary $pl
}
```

 # setting ignored layers

```
set_ignored_layers -min_routing_layer M2 -max_routing_layer M6
```

 # fix the macros

```
set_fixed_objects [get_flat_cells -filter "is_hard_macro"]
```

 # checking_for_congestion

```
create_placement -floorplan
```

 # legalized_placement

```
legalize_placement -incremental
```

 # define report directory

```
set rpt_dir "./reports/FLOORPLAN"
# create directory if it does not exist
if {![file exists $rpt_dir]} {
   file mkdir $rpt_dir
}
```

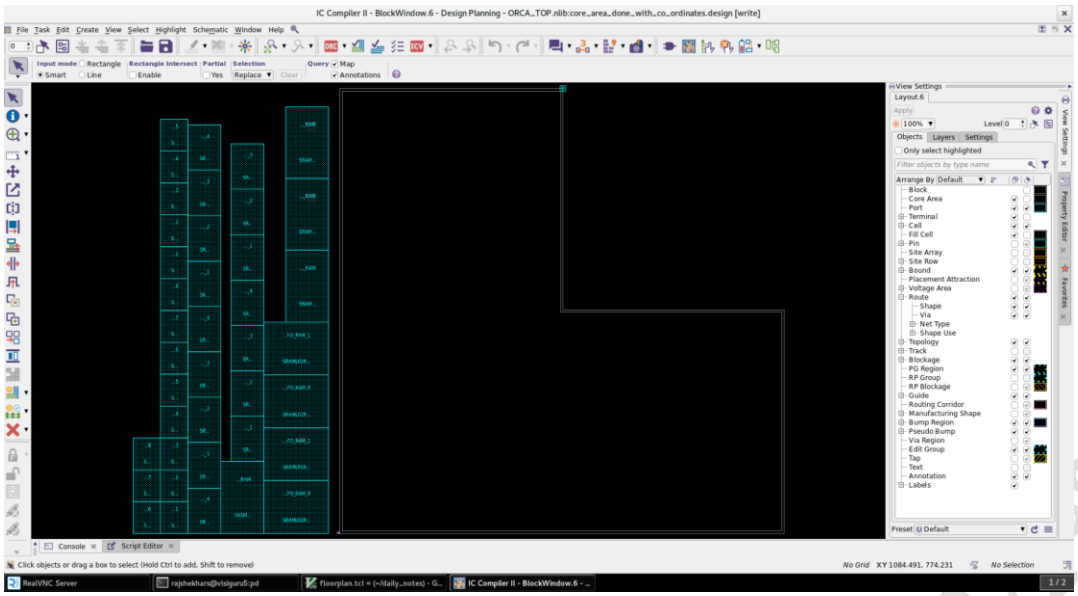
 # congestion report

```
redirect ./reports/FLOORPLAN/fp_congestion.txt {report_congestion -rerun_global_router}
```

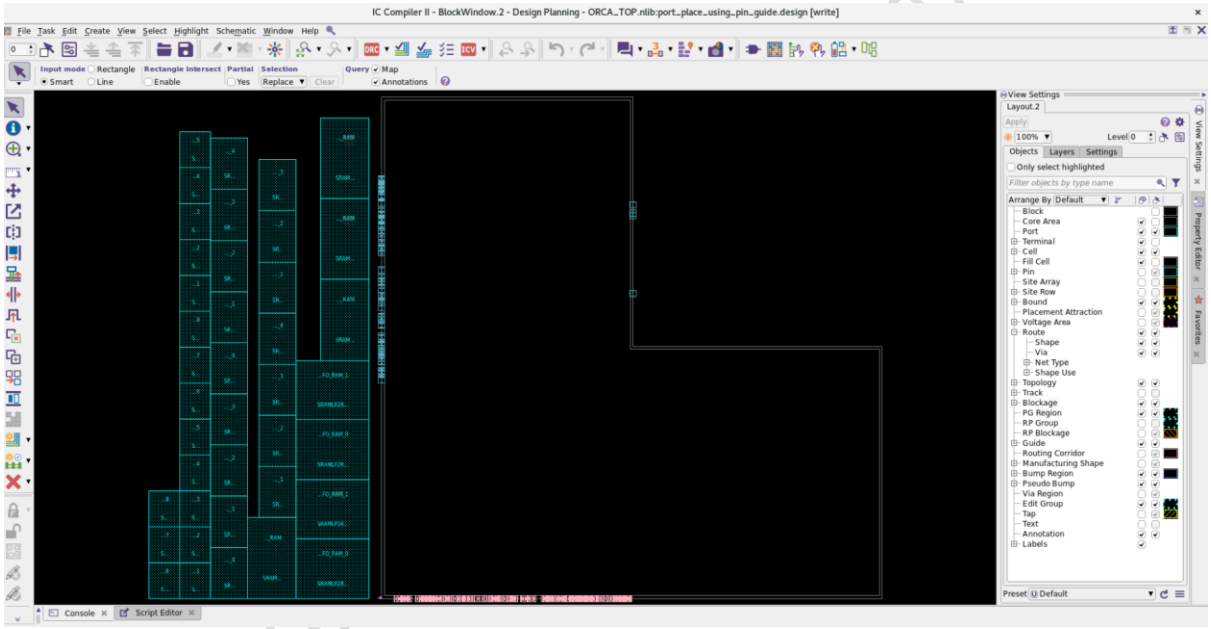
 # reset the placement

```
reset_placement
# boundary cells
get_lib_cells -nocase *dcap_hvt*
set b_cell [get_lib_cells -nocase *dcap_hvt*]
set_boundary_cell_rules \
  -left_boundary_cell  $b_cell \
  -right_boundary_cell $b_cell \
  -at_va_boundary
compile_boundary_cells
check_boundary_cells
##Tap cells
create_tap_cells -distance 30 -pattern stagger -lib_cell $b_cell -skip_fixed_cells
check_legality
# checks reports
check_boundary_cells      > ./reports/FLOORPLAN/boundary_cell.rpt
check_legality            > ./reports/FLOORPLAN/TAP_cell.rpt
check_physical_constraints > ./reports/FLOORPLAN/physical_constr.rpt
check_pin_placement -wire_track true > ./reports/FLOORPLAN/pin_placement.rpt
# saving the block
save_block -as floorplan
#view in gui
Start_gui
```

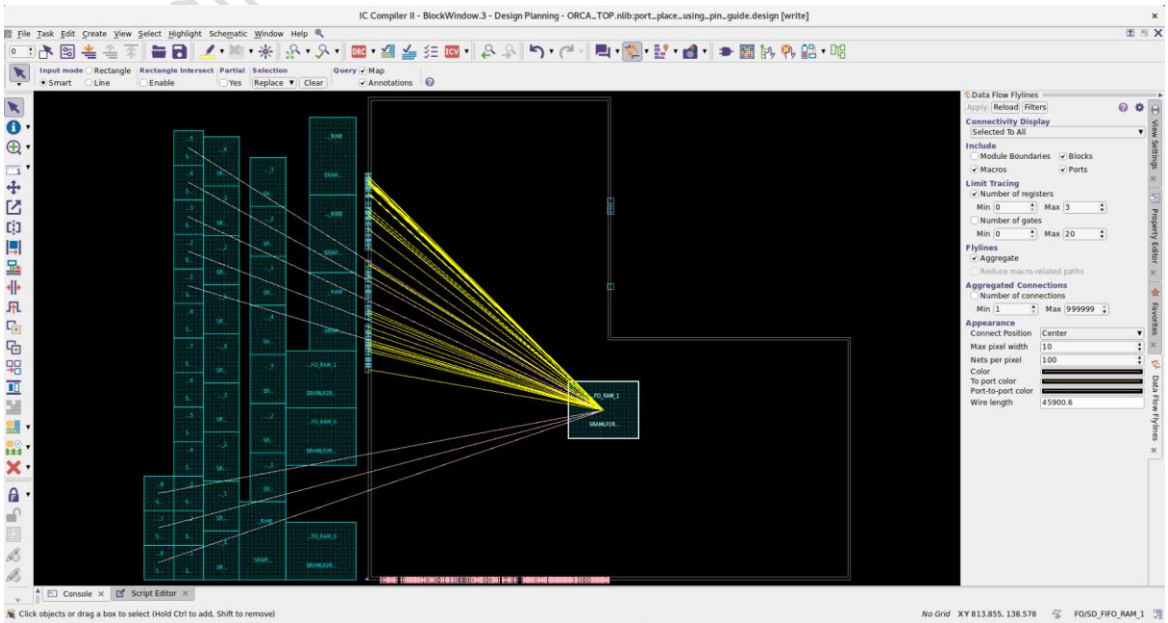
#Core_area and Die area



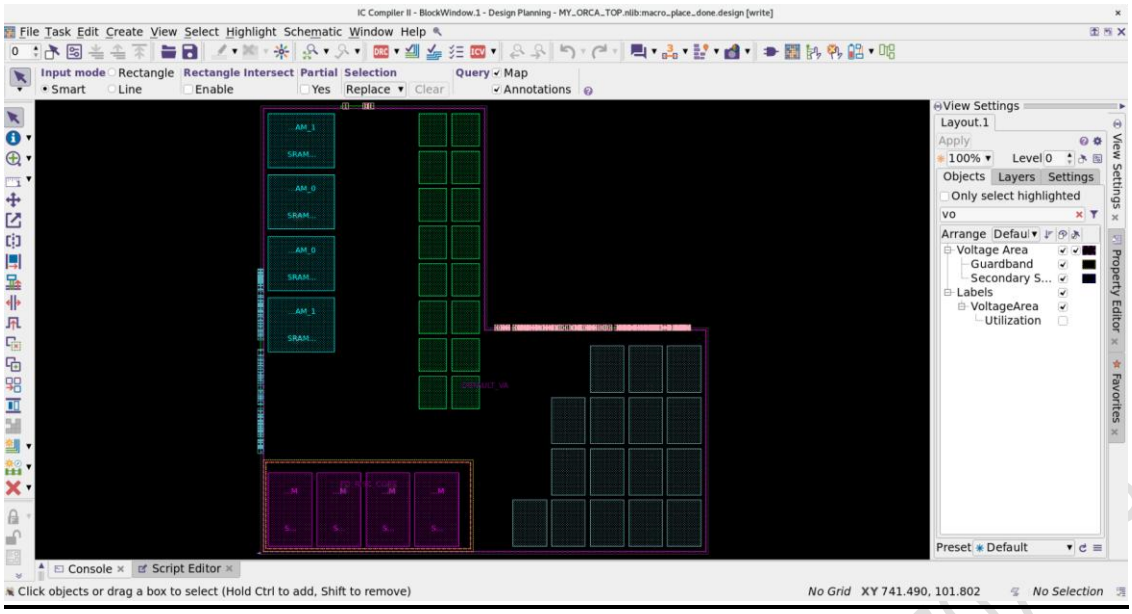
#Port_placement



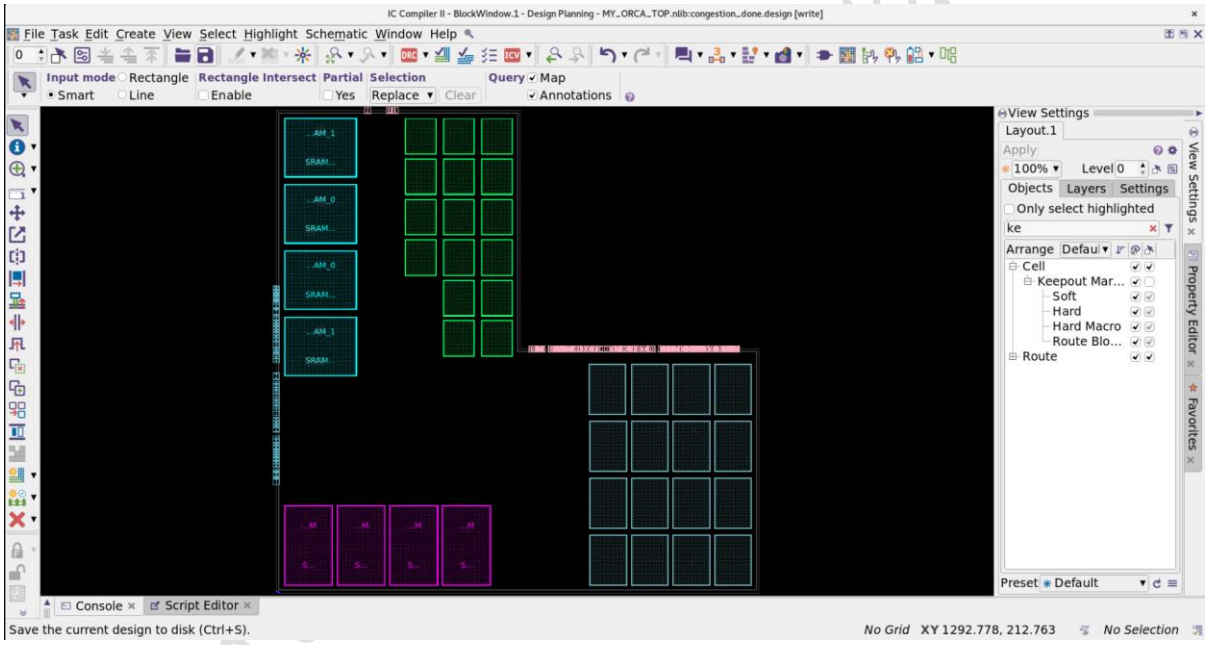
#Macro placement based on flyline analysis



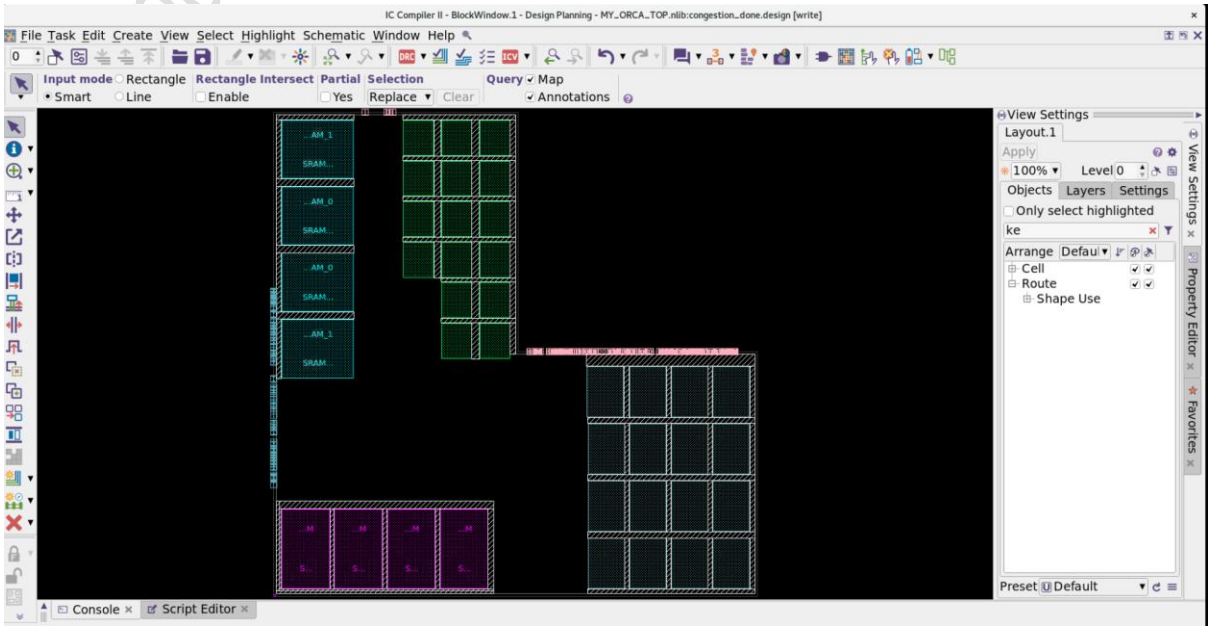
#Voltage area creation



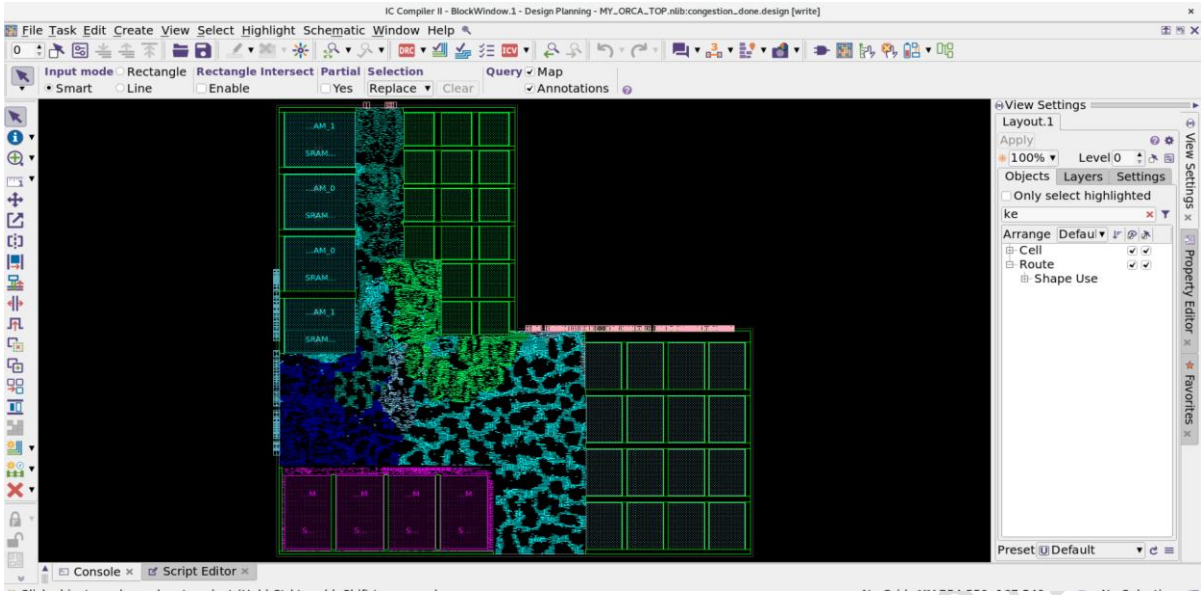
#Keepout_margin



#Creating partial blockage



#Congestion report




```
****************************************
Report : congestion
Design : ORCA_TOP
Version: S-2021.06-SP2
Date   : Sat Oct 18 23:58:12 2025
****************************************

Layer      |    overflow     |              # GRCs has
Name       |  total  |  max  | overflow (%)       | max overflow
-------------------------------------------------------------------
Both Dirs  |      48 |     1 |      48  ( 0.01%)  |      48
H routing  |      48 |     1 |      48  ( 0.02%)  |      48
V routing  |       0 |     0 |       0  ( 0.00%)  |       0

1
```