

**# ICC2 : Power Planning Automated Script**

**# Tool : Synopsys IC Compiler II (ICC2)**

**# Stage : Power Planning**

**# Date : 25-01-2026**

**# Sanity check: power nets**

```
set pwr_nets [get_nets -quiet {VDD VSS VDDH}]

if {[sizeof_collection $pwr_nets] == 0}{

    puts "ERROR: Power nets not found. UPF may not be loaded."

    return

}
```

**# Open floorplan block and create powerplan block**

```
open_lib ORCA_TOP.nlib

copy_block -from_block floorplan -to_block powerplan

open_block powerplan
```

**# Create report directory**

```
set rpt_dir "./reports/POWERPLAN"

if {! [file exists $rpt_dir]}{

    file mkdir $rpt_dir

}

If 0 {
```

#### **Source power planning implementation**

“The power planning script is a reusable template derived from a previous project. I adapted it to this design by modifying layers, widths, spacing, ring strategy, and checks based on current requirements.”

```
}
```

**#create\_pg\_pattern : Physical aspects . spacing pitch offset direction**

**# create\_pg\_strategy : How to use pattern in design**

**# set Via rule**

**# compile strategy and via rule : physical stuctures are created**

**remove\_pg\_strategies -all**

**remove\_pg\_patterns -all**

```
remove_pg_regions -all

remove_pg_via_master_rules -all

remove_pg_strategy_via_rules -all

remove_routes -net_types {power ground} -ring -stripe -macro_pin_connect -
lib_cell_pin_connect

# update the connections

connect_pg_net

# via rule

set_pg_via_master_rule pgvia_8x10 -via_array_dimension {8 10}

# all macros

set all_macros [get_flat_cells -filter "is_hard_macro && !is_physical_only"]

# macros belongs to I_RISC_CORE

set hm(risc_core) [get_flat_cells -filter "is_hard_macro" I_RISC_CORE/*]

# macros belongs to PD_ORCA TOP power domain

set hm(top) [remove_from_collection $all_macros $hm(risc_core)]

##### create pattern, strategy for higher straps M7, M8 #####
#####

create_pg_mesh_pattern P_top_two \
-layers { \
    {horizontal_layer: M7} {width: 1.104} {spacing: interleaving} {pitch: 13.376} {offset: 0.856} \
    {trim: true} } \
    {vertical_layer: M8} {width: 4.64} {spacing: interleaving} {pitch: 19.456} {offset: 6.08} {trim: \
    true} } \
} \
-via_rule { \
    {intersection: adjacent} {via_master : pgvia_8x10} \
}

# strategy

# for core vdd, vss

set_pg_strategy S_default_vddvss \
-core { \
-pattern { \
    {name: P_top_two} {nets:{VSS VDD}} {offset_start: {0 0}} \
}
```

```

}\\
-blockage { \
{nets: VDD} {voltage_areas: PD_RISC_CORE}} \
}\
-extension { \
{stop:design_boundary_and_generate_pin}} \
}\
}

# for voltage area -VDDH

set_pg_strategy S_va_vddh -voltage_areas PD_RISC_CORE \
-pattern { \
{name: P_top_two} {nets:{- VDDH}} {offset_start: {0 0}} } \
}\
-extension { \
{direction:BR}{stop:design_boundary_and_generate_pin}} \
}\
compile_pg -strategies {S_va_vddh S_default_vddvss}

#####
##### create pattern , strategy , via rule for lower strap M2
#####

create_pg_mesh_pattern P_m2_triple \
-layers { \
{vertical_layer: M2} \
{track_alignment : track} \
{width: {4.04 0.192 0.192}} \
{spacing: {2.724 3.456}} \
{pitch: 9.728} {offset: 1.216} \
{trim : true} \
}
set_pg_strategy S_m2_vddvss \
-core { \
-pattern { \
{name: P_m2_triple} {nets: {VDD VSS VSS}} {offset_start: {0 0}} } \
}

```

```

}\\
    -blockage { {nets: VDD}{voltage_areas: PD_RISC_CORE}{macros_with_keepout:
$all_macros} \
    -extension {{stop:keep_floating_wire_pieces}} \
}

set_pg_strategy S_m2_vddh \
    -voltage_areas PD_RISC_CORE \
    -pattern { \
        {name: P_m2_triple} {nets: {VDDH - .}} {offset_start: {0 0}} \
    } \
    -blockage { {macros_with_keepout: $hm(risc_core)} } \
    -extension {{direction:B} {stop:design_boundary_and_generate_pin}} \
# Via rules

set_pg_strategy_via_rule S_via_m2_m7 \
    -via_rule { \
        {{strategies: {S_m2_vddvss S_m2_vddh}} {layers: {M2}} {nets: {VDD VDDH}}} \
        {{strategies: {S_default_vddvss S_va_vddh}} {layers: {M7}} } {via_master: {default}} \
        {{strategies: {S_m2_vddvss S_m2_vddh}} {layers: {M2}} {nets: {VSS}}} \
        {{strategies: {S_default_vddvss S_va_vddh}} {layers: {M7}} } {via_master: {default}} \
    }
#####
##### compile the strategies and via rules
#####

compile_pg -strategies {S_va_vddh S_m2_vddh}
compile_pg -strategies {S_default_vddvss S_m2_vddvss} -via_rule {S_via_m2_m7}
#####
##### Build rings around the macros then connect them
#####

suppress_message PGR-599
#####
##### create pattern, strategy , via_rule for macro rings
#####

create_pg_ring_pattern MACRO_RING_VDD_PATTERN \
    -horizontal_layer M5 \
    -vertical_layer M6 \
    -width 0.5 \

```

```
-spacing 0.5

set_pg_strategy MACRO_RING_VDD_STRAGEY \
set hm(top)

-pattern {name: MACRO_RING_VDD_PATTERN}{nets: {VDD VSS}}{offset: {0.3 0.3}}}

create_pg_ring_pattern MACRO_RING_VDDH_PATTERN \
    -horizontal_layer M5 \
    -horizontal_width 0.5 \
    -vertical_layer M6 \
    -vertical_width 0.5

set_pg_strategy MACRO_RING_VDDH_STRAGEY \
    -macros $hm(risc_core) \
    -pattern {name: MACRO_RING_VDDH_PATTERN}{nets: {VDDH VSS}}{offset: {0.3 0.3} }

set_pg_strategy_via_rule S_ring_vias \
    -via_rule {
        {{strategies: {MACRO_RING_VDD_STRAGEY MACRO_RING_VDDH_STRAGEY}}{layers: {M5}}}\

        {existing: {strap }}{via_master: {default}} \
        {{strategies: {MACRO_RING_VDD_STRAGEY MACRO_RING_VDDH_STRAGEY}}{layers: {M6}}}\

        {existing: {strap }}{via_master: {default}} \
    }

### compile strategies

compile_pg -strategies \
    {MACRO_RING_VDD_STRAGEY MACRO_RING_VDDH_STRAGEY} \
    -via_rule S_ring_vias

##### create pattern, strategies and compile for macropins
#####

create_pg_macro_conn_pattern P_HM_pin \
    -pin_conn_type scattered_pin \
    -layers {M5 M6}

set_pg_strategy S_HM_top_pins \
    -macros $hm(top) \
    -pattern { {pattern: P_HM_pin}{nets: {VSS VDD}} }
```

```

set_pg_strategy S_HM_risc_pins \
    -macros $hm(risc_core) \
    -pattern { {pattern: P_HM_pin} {nets: {VSS VDD}} } }

compile_pg -strategies {S_HM_top_pins S_HM_risc_pins}

##### Build the standard cell rails #####
create_pg_std_cell_conn_pattern P_std_cell_rail

set_pg_strategy S_std_cell_rail_VSS_VDD \
    -core \
    -blockage { {nets: {VDD} {voltage_areas: PD_RISC_CORE}} {macros_with_keepout: $all_macros} } \
    -pattern { {pattern: P_std_cell_rail} {nets: {VSS VDD}} } \
    -extension {{stop: outermost_ring}{direction: L R }} \

set_pg_strategy S_std_cell_rail_VDDH \
    -voltage_areas PD_RISC_CORE \
    -blockage {macros_with_keepout: $all_macros} \
    -pattern {{pattern: P_std_cell_rail}{nets: {VDDH}}}

set_pg_strategy via_rule S_via_stdcellrail \
    -via_rule {{intersection: adjacent}{via_master: default} }

compile_pg -strategies {$std_cell_rail_VSS_VDD $std_cell_rail_VDDH} \
    -via_rule {$via_stdcellrail}

# Power connectivity & DRC checks

check_pg_connectivity -check_std_cell_pins none > $rpt_dir pg_connect.rpt
check_pg_missing_vias > $rpt_dir pg_miss_vias.rpt
check_pg_drc -ignore_std_cells > $rpt_dir pg_drc.rpt

# Save power planned block

save_block -as powerplan

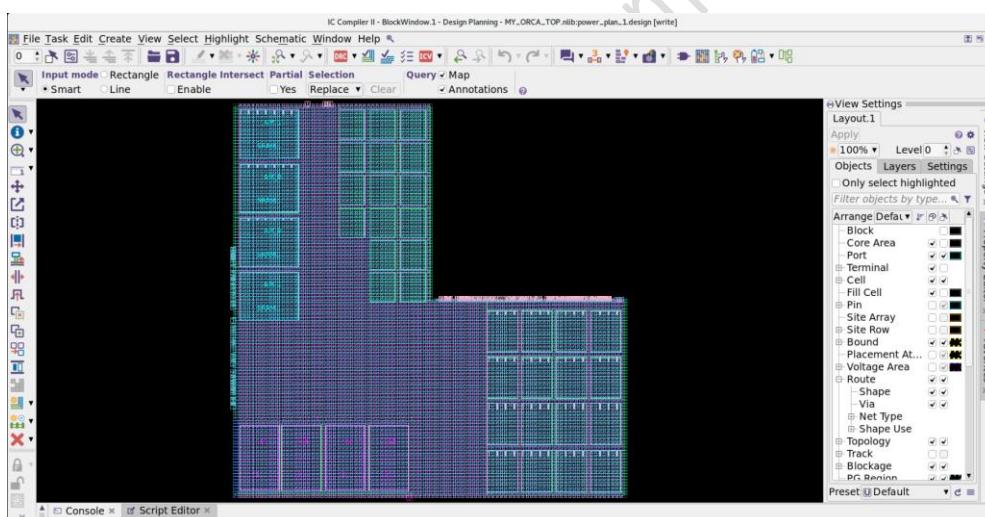
```

## FIXES

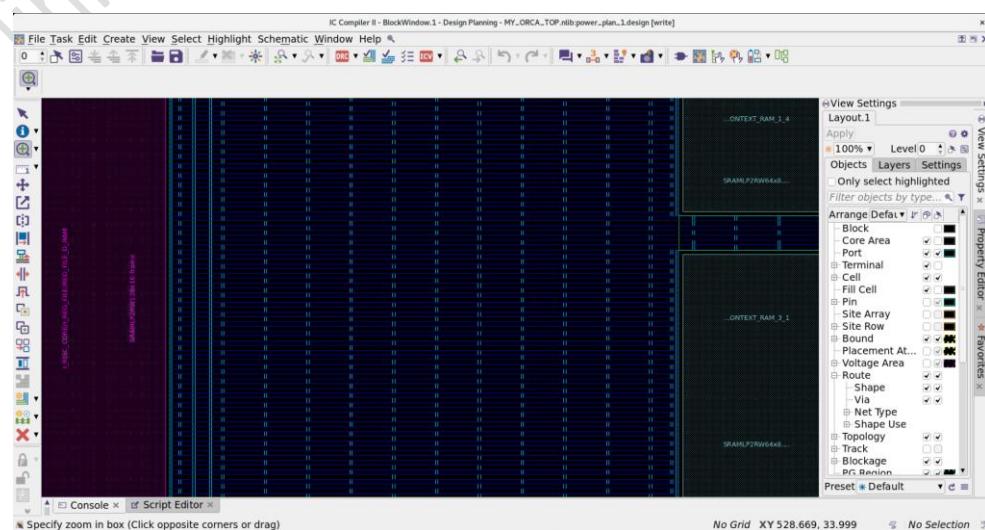
- Floating wires, missing vias, and routing DRC violations are observed due to improper macro placement and limited routing space.
- Macros are repositioned to create sufficient routing channels for both signal and power nets.
- Adequate spacing is maintained between adjacent macros to allow smooth VDD and VSS strap routing.
- Macro alignment is optimized to avoid blocking power ring and stripe continuity.
- Routing blockages are applied around macros to control routing direction and reduce congestion.
- Improved macro placement enables proper via insertion across multiple metal layers.
- Power nets are ensured to pass through macro channels without obstruction or discontinuity.
- These adjustments eliminate floating wires and missing via violations.
- Proper macro placement reduces IR drop and improves overall power integrity.
- The design achieves DRC-clean routing with stable and reliable power delivery.

#to view gui

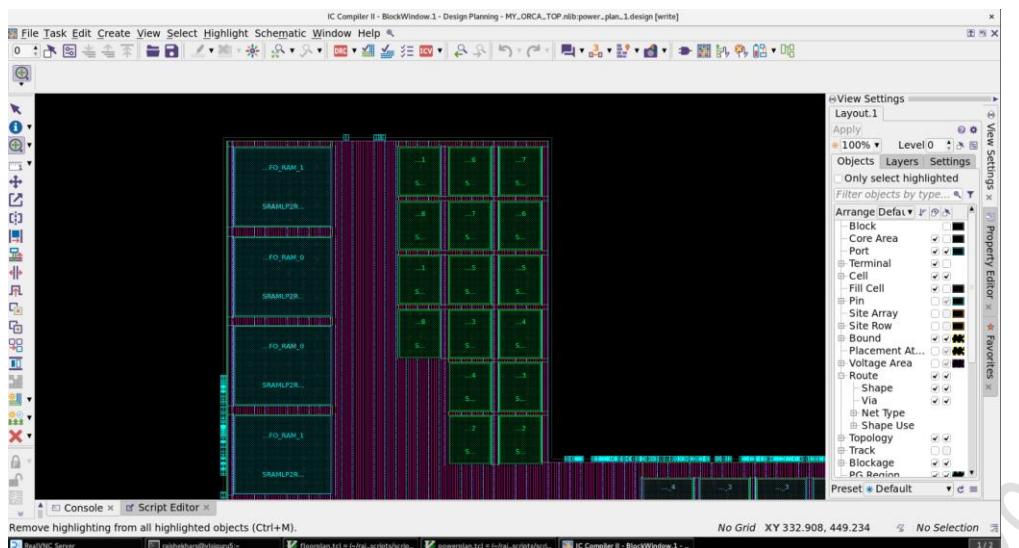
Start\_gui



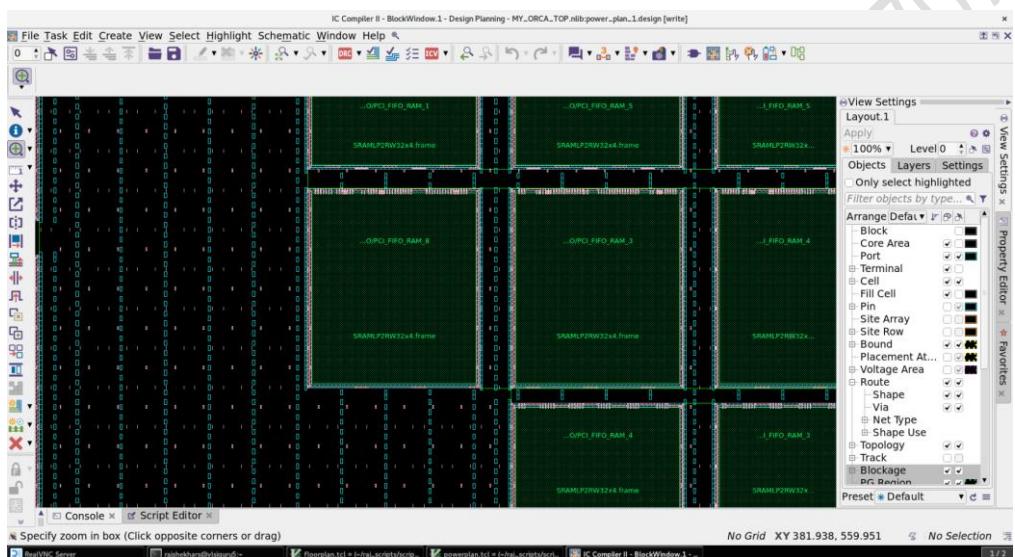
M1 RAIL



M2 STRAP



## M5&M6 MACRO RINGS



## M7 & M8 POWER MESH

