International Journal of Advanced Technology and Engineering Exploration, Vol 8(80)
ISSN (Print): 2394-5443 ISSN (Online): 2394-7454
http://dx.doi.org/10.19101/IJATEE.2021.874162

Research Article

PNR flow methodology for congestion optimization using different macro placement strategies of DDR memories

J. Fadnavis[1*] and Kariyappa B.S[2]
Student, Electronics and Communication Engineering Department, RV College of Engineering, Bangalore, India[1]
Professor, Electronics and Communication Engineering Department, RV College of Engineering, Bangalore, India[2]

Received: 29-May-2021; Revised: 15-July-2021; Accepted: 18-July-2021

Abstract
The demand for high-performance electronic gadgets has increased two-folds in the last decade, fueling technology manufacturers to shrink fabrication node sizes. The decreasing channel sizes along with an increase in gate count and cell density pose numerous congestion issues during physical implementation of the chips, making design closure ever more difficult. Double Data Rate (DDR) memories that access data on both edges of the clock cycle require extreme timing control and must meet the strict timing requirements during Physical Design (PD). Floor-plan, being the first stage of back-end PD implementation, is an important step to mitigate congestion and timing issues during the subsequent stages of the implementation. On-chip macros, with connections to the standard cells and the Input/Output (IO) ports of the chip, need to be strategically placed during the floor-plan of the design to enable congestion-free placement of standard cells and signal routes. Previously, designers opted for island macro placement strategy, wherein macros were grouped close together, thereby leaving a uniform square region for standard cell placement. However, this method alone cannot be considered for chip designs today that has denser macro pin connections to the chip IO ports as in the Last Level Cache (LLC) block of a DDR subsystem. In this paper, two new placement strategies have been considered – peripheral and donut, for the LLC module. A congestion-optimized, floor-plan to Place and Route (PNR) flow methodology has been presented for each of these placement strategies using Cadence Innovus Implementation System and Synopsis IC Compiler II. The Quality of Results (QOR) for each strategy was then compared. The peripheral macro placement strategy is found to be best among the three, while the donut macro placement is the worst. A 16% improvement in the overall on-chip delay is seen in the peripheral macro placement when compared to island macro placement. Furthermore, a 19.6% power reduction is observed in the peripheral macro placement strategy as compared to island macro placement. The overall congestion for peripheral macro placement is 0.32%, which is the least among the three strategies. Hence, the peripheral macro placement strategy proves to be the best choice for macro placement, when considering floor-plan for the LLC module in a DDR subsystem.

Keywords
Double data rate, Physical design, Floor-plan, Macro placement, Island, Peripheral, Donut, Congestion.

1.Introduction
The large-scale growth of the electronics industry in the past decade has led to small-sized gadgets flooding the market. Initially, only Personal Computers (PC) employed the use of multi-core processing, however, now even small handheld gadgets have come to possess similar computational power as the PC and laptop. With mobile manufacturers opting for multi-core processing and graphics-intensive application usage, memory storage and access has become a crucial aspect to achieve high performance. In recent years, the memory subsystem is viewed as a bottleneck; drastically reducing performance speeds. Hence, there has been a unanimous shift towards the Double Data Rate (DDR) memory subsystems. The DDR acts as an interface between the controller and memory by providing data access on both edges of the clock cycle, thereby, doubling the memory access speed to accommodate the growing needs of computationally extensive tasks.

The DDR subsystem has been increasingly employed even in the Internet of Things (IoT) sector. There is a great problem of reliability and efficient resource management concerning the sensors and actuators that are used for real-time data gathering and local preprocessing [1]. The size of wearable devices has shrunk drastically, due to the advancement of semiconductor technology, which has fuelled the

*Author for correspondence

usage of wearable sensors and smart devices [2]. The memory requirements for such devices are stringent due to real-time data access requirements, imposing strict timing constraints. The usage of large-scale access control lists in IoT applications [3], furthers the need for high-speed data access solutions.

The DDR memory subsystem consists of the controller, physical interface, and Input/Output (I/O) drivers. One of the important blocks within the DDR is the Last Level Cache (LLC). The LLC is a standalone memory inserted between the external memory and functional blocks to provide another level of cache. The LLC is the last memory level to be checked on-chip before moving to fetch data from external memory. The time to access data from an off-chip memory is very high, hence the LLC, acting as a buffer cache, helps to reduce data fetch off-chip. Since the DDR is at the interface of the chip and off-chip memory, extreme timing control is required to ensure the correct functioning of DDR, requiring several hardware components and algorithms to facilitate this complex design. Numerous architectural optimizations such as deep pipelines, branch prediction, and aggressive reordering aim to provide high performance [4]. The substantial research carried out to improve the efficiency of this subsystem, has increased the gate level complexity and power consumption of this subsystem. A 33% power consumption of the LLC and Dynamic Random-Access Memory (DRAM) alone is observed in the DDR subsystem [5]. The high gate density and critical timing due to the physical interface with the off-chip memory needs to be taken care of at the PD implementation level.

Physical Design (PD) implementation is a back-end flow from the net-list to Graphic Data Stream (GDS) and is the correlation step between design and chip manufacture. The PD flow ensures that the design created works on the silicon chip. Numerous problems arise when the design is converted to one, which can function properly on silicon. The PD flow first involves proper planning and placement of pins, and custom macros on the chip during the floor-plan stage. Next, the placement of the logic on the chip is performed along with the introduction of the clock and power distribution network. The design is then routed and checked for various parameters such as power, area, and performance. The congestion and timing requirements need to be met during PD to facilitate the correct functioning of the design. All blocks of the chip need to be implemented and tested according to this PD implementation flow. As the Very Large-Scale Integration (VLSI) system's multifaceted nature increments instantly, physical planning is getting increasingly troublesome [6].

Various floors-plan techniques have been explored in the past, such as a partition level floor-plan method to understand the in-depth structure of the block to decide floor-plan and obtain better timing [7].

Challenges such as large design sizes, increasing macro count, timing/power estimations, region shaping and pin assignment, predefined placement locations, macro-orientations, and pin positions, simultaneous standard cell and macro placement, congestion, and timing-driven placement is increasing for a floor-plan designer [8].

Macro placement is a crucial step to obtain congestion-free designs at the later stages of PD flow. The placement of standard cells, which is done by the placement tool, ideally requires a uniform square region on-chip, to perform an optimum placement. To satisfy this requirement, designers initially employed the island macro placement configuration for a floor-plan designs, which groups all macros in one corner of the chip to provide such a uniform region for standard cells. However, as the macro pin connection to IO ports of the chip grows denser, this method is inefficient and often leads to more congested designs. Therefore, there is a need to explore different macro placement strategies to avoid such congested designs. In this paper, two new macro placement strategies, peripheral macro placement, and donut macro placement have been explored for the LLC block, and complete congestion optimized Place and Route (PNR) flow for each of these has been implemented using Cadence Innovus Implementation Systems and Synopsis IC Compiler II. The various inbuilt settings of these powerful tools have been leveraged to optimize congestion and improve timing integrity throughout the PNR flow. The Quality of Results (QOR) of the three macro placement strategies was then compared to arrive at the best choice for macro placement for LLC modules in DDR subsystems.

2.Literature review

A detailed study of state-of-the-art architectures of DDR and LLC was carried out. Various developments in the architecture of both blocks have been explored in the past decade to reduce the latency and power of each. These developments give an idea to appreciate the complexity involved during the PD process to ensure the proper functioning of the blocks. Numerous floor-plan techniques have also been developed in the

past to improve congestion and timing of the blocks during back-end flow.

A Power Delivery Network (PDN) in a Re-Distribution Layer (RDL) for a DDR memory subsystem is presented in [9]. It is observed that decreasing the PDN loop inductance is critical for a robust high-speed PDN design. The loop inductance depends on the wire width and length. Hence, the wire parameters used for the power distribution network must be altered. A voltage ripple reduction between the Power/Ground (PG) rails is done by a simple PDN model. The voltage ripple reduction is caused by opting for symmetric PG PDN structure and a unity PG ratio is a must for maintaining the power integrity of the design along with keeping signal integrity at an optimum level. The DDR memory controller is the brain of the entire DDR subsystem, hence, an optimized controller design as discussed in [10] can improve the performance of the overall subsystem. All commands like read/write access and pre-charge commands were tested and verified. The verification was done on System Verilog to provide high coverage for the code to make sure the perfect functioning of the block. The controller was designed to generate timing and control signals to synchronize the command operations. The drawback of the above design is an increased number of buffers that are inserted. The inserted buffers result in an extra delay in the data paths, which severely affect the timing closure of the designs.

The power rail noise limit is determined by the DDR to interface current profile and PDN impedance [11]. The dynamic behavior of the memory subsystem greatly increases the power rail noise due to the sudden charge and discharge of current through the Static Random-Access Memory (SRAM) cells. On-die PDN is studied using the solution space analysis, wherein the power rails are decomposed into lumped on-die capacitors and effective series resistance. Different currents and voltages are applied to emulate the various operating conditions to estimate the overall voltage drop. The analysis shows that higher capacitance and low series resistance lowers the voltage drop. A design of freeway Network On-Chip (NoC) is proposed in [12] which routes flits on DDR and allows bypass pipelining. Pipeline bypassing reduces the packet latency at a low traffic load. The routing is done in such a way that only flits moving straight can pass through the bypass pipeline. In smaller networks, the freeway latency is found to be 49% higher than short-path, but in large networks, the freeway-NoC latency is 5% lower with a 23% increase in throughput. A sensitivity analysis to estimate the signal and power integrity of a PDN for a DDR is presented in [13]. A synthesized Resistance-Inductance-Capacitance (RLC) model is proposed to perform model extraction instead of the Computer-Aided Design (CAD) layout model extraction. The model is created using self and transfer impedance equations that can be incorporated into an algorithm. The models are created quickly and efficiently and match very closely to the CAD layout extraction models. The models are passive and causal, and correlation is good for both frequency and time domains. The above method produces faster analysis results while maintaining the accuracy of the CAD layouts.

With advancements in DDR, rapid advancement in LLC technology has also taken place to keep up with fast-growing memory needs. Several developments in LLC have taken place in the past decade. One such development is in stacking technology. The increased parallelism in LLC has resulted in opting for 3D stacking as compared to the traditional 2D stacking. However, the leakage power is seen to increase greatly due to dense 3D integration [14]. A novel hybrid reconfigurable architecture for LLC was proposed. The new design combines SRAM along with Spin Transfer Torque (STT) SRAM technology to dynamically reduce power at runtime by restoration and duplication. The power is seen to reduce by 98.4% as compared to the traditional design. A cache-partitioning algorithm is used to efficiently divide the LLC block among the different processors. A novel method to partition cache using Non-Volatile Memories (NVM) instead of SRAM is presented in [15]. The cache is periodically portioned in such a way, to assign heavily accessed ways to low accessed partitions, thereby distributing the access to the entire LLC block.

Sakhare et al. [16] presented the replacement of SRAM-based LLC with STT Magnetic Random Access Memory (MRAM) based LLC, due to the limited scaling capability of SRAM. The STT-MRAM based design proves to provide larger energy gains and low access latency. Two more designs, Compressed Tag (CT) cache [17] and data shepherding [18] are presented to manage larger LLC blocks. The developments on DDR and LLC have increased the cell level complexity and timing criticality during PD flow. The larger number of logic cells that was inserted to optimize the design in terms of power, results in a highly congested placement if care is not taken to prepare the floor-plan. Several floor-plan and macro

placement techniques have been explored to enable congestion-free standard cell placements and overcome the higher logic density challenges.

A macro placement algorithm for regular placement of macros is presented in [19]. Macros and standard cells are clustered together in advance according to the connections between them, creating different hierarchies of macros. The macros are then legalized to obtain an efficient floor-plan. The simulated annealing algorithm combined with the corner stitching algorithm is explored for macro placement in [20]. This method is effective to refine the placement of standard cells along with macros according to the placement regions defined by the algorithm. A clustering algorithm for standard cells and macros built as a tree from the design hierarchy during synthesis is presented in [21], allowing the algorithm to consider the indirect connectivity of macros to the standard cells. This method is best used when the placement of macros and standard cells is done simultaneously.

A novel multi-level algorithm that considers the Register Transfer Logic (RTL) connections between macros and standard cells is discussed in [22]. The synthesis net-list is divided based on dataflow hierarchy and a cost function is evaluated to optimize the wire length and timing of the connections. The proposed algorithm enables easy timing and Design Rule Check (DRC) closure. The amount of impact that the macro placement has on the congestion of the design is assessed in [23]. Two different macro placement strategies take into consideration and the impact at each PD stage is evaluated. The congestion and QOR are observed at every step to assess the effect of macro placement. To ensure the timing closure of the design, several manual optimizations are required to meet the setup and hold times. Different methods to fix the setup and hold time are given in [24]. These methods provide a robust timing closure method to obtain minimal DRCs during the sign-off phase of PD implementation. Various algorithms exist to group and slice up the cell into the gate level net-list according to parameters such as maximum interconnect length and logical depth. One such algorithm is the Genetic and Simulated Annealing (GSA) algorithm [25], which is used to define weight values for different cells while clustering them to perform an efficient placement. The macro placement has further been explored as a fully automated solution using machine learning models in [26]. Several floors-plans with different macro placements have been provided to build a robust machine learning model to decide the optimum macro placement for giving floor-plan specifications.

The above macro placement techniques, while accounting for the connections between macros and standard cells, do not account for the connections between macros and I/O ports of the chip. With modules such as the LLC, the macros majorly have connections to the I/O ports of the chip. These connections are of utmost importance in an LLC module, as these ultimately interfaces with the off-chip memory. Moreover, the above algorithms are provided for a full-custom PNR flow, where macros and standard cells are simultaneously placed, which requires such automated algorithms. However, since the LLC module is developed as a semi-custom design, the macros are placed first, followed by standard cells. In this paper, two macro placement strategies are presented for semi-custom flow that takes into consideration the connections of macros to the standard cells as well as the I/O ports of the chip.

3.Methods

3.1DDR subsystem

The System-on-Chip (SoC) design needs to interface with the off-chip memory as shown in *Figure 1*. The memory subsystem is shared and must respond to numerous requests from multiple cores, each having its latency and bandwidth requirements. The processor, along with the Graphics Processing Unit (GPU) and Digital Signal Processor (DSP), interacts with the memory. To decrease the memory access times, the DDR subsystem acts as an interface between the processors and the memory. The DDR enables memory access on both edges of the clock cycle as compared to the traditional memory systems accessing data on only one clock edge. One of the blocks in the DDR subsystem is the LLC. The LLC acts as an additional cache memory apart from the L1 and L2 caches. The LLC was added as an attempt to further reduce the memory access times by reducing the frequency of data access that is off-chip.

The DDR subsystem has been increasingly employed in applications such as satellite navigation [27]. However, the physical interface and high-speed data access, impose tight PD constraints on the module. Such new architecture furthers the need to meet timing requirements in all extreme corner cases to ensure the proper functioning of the memory interface across several environmental conditions. Hence, a detailed PD implementation is required to ensure the working of this subsystem.

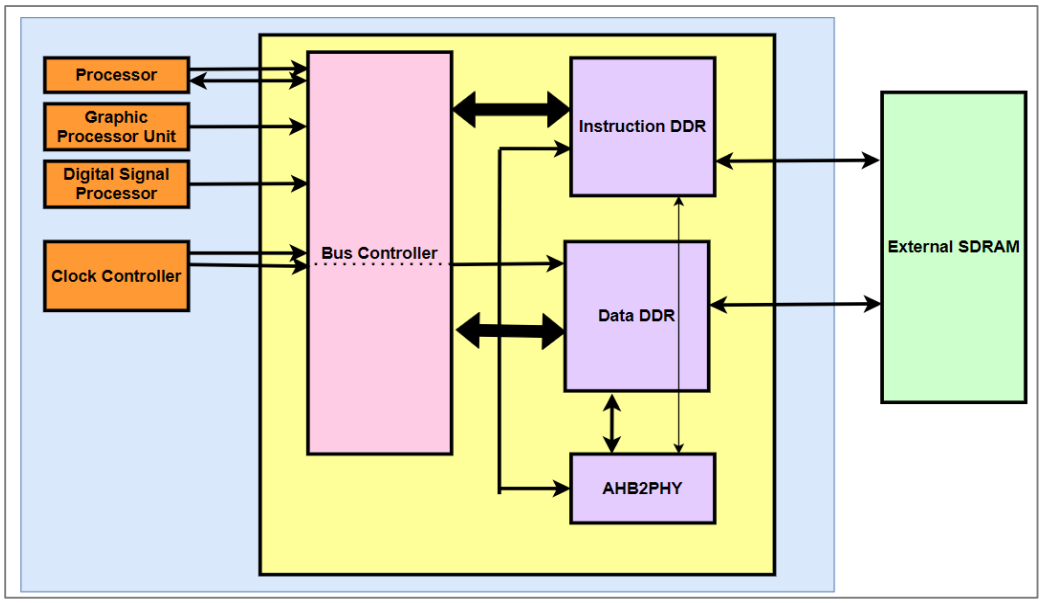

Figure 1 SoC Architecture

3.2Overall methodology

The PD implementation starts with importing the gate level net-list file, Synopsis Design Constraints (SDC) file, physical and technology files, timing liberty modules, and power intent file. The back-end flow, then begins from the floor-plan stage. The first step is to define the core utilization and the aspect ratio as seen in *Figure 2*. The core to IO boundary is then decided to snap the corners of the instance grid. Next, the pin placement on the boundary is done based on the inputs from the top-level hierarchy module, and appropriate layers are assigned to the pins. Macro placement is carried out using the three different strategies – island, peripheral, and donut. The various placement blockages are then defined to ensure the cleanliness and congestion-free placement of standard cells. Placement regions are further defined grouping similar logic hierarchy cells together. The physical-only cells are then placed all over the core area. Finally, power rings and straps are generated based on the power intent of the design. A sanity check on the floor-plan is performed to ensure a clean design before placement.

The standard cell placement is performed by the tool using several inbuilt algorithms. The placement begins with the initial coarse placement that places the standard cells randomly according to the space available. This information is then used to perform optimization, to adjust cells to reduce congestion and meet timing. The next step is the refine incremental placement, wherein small perturbations are carried out iteration-by-iteration to optimize the design further. The final placement and optimization, then take place, followed by the legalization to snap to the manufacturing grid. Next, the clock specifications need to be defined to lay out the clock network on the chip. To form the multi-clock tree, clock drivers are created followed by clock straps generation. Once the clock mesh is ready, the global clock tree is built and checked. Next, the clock mesh is routed and the entire clock tree is synthesized and legalized.

The routing begins with routing clock and certain critical nets. Next, the secondary power grid mesh is connected. The global route is then performed, where approximate routes are assigned and coarse congestion is calculated. The track assignment is the step where the tracks of different routing layers are assigned to the global routes. After the track assignment, violations may exist which are resolved during the detailed placement stage. Post-route optimization is performed to fix congestion and legalize the routing. Once routing is complete, the sign-off checks consisting of timing, congestion, area, and power analysis are performed. The setup and hold timings are fixed based on timing reports generated, by size or replacing buffers and inverters. The DRC checks are performed to make sure the design is ready for manufacture and involves metal filling and Engineering Change Order (ECO) fixes.

The above methodology is followed for each of the three different macro placement strategies along with leveraging the various tool options provided by Synopsis IC Compiler II, which are employed to

implement power and performance-optimized design for each. The timing, power, and congestion values for each of the macro placement strategies were observed and analyzed at each stage.

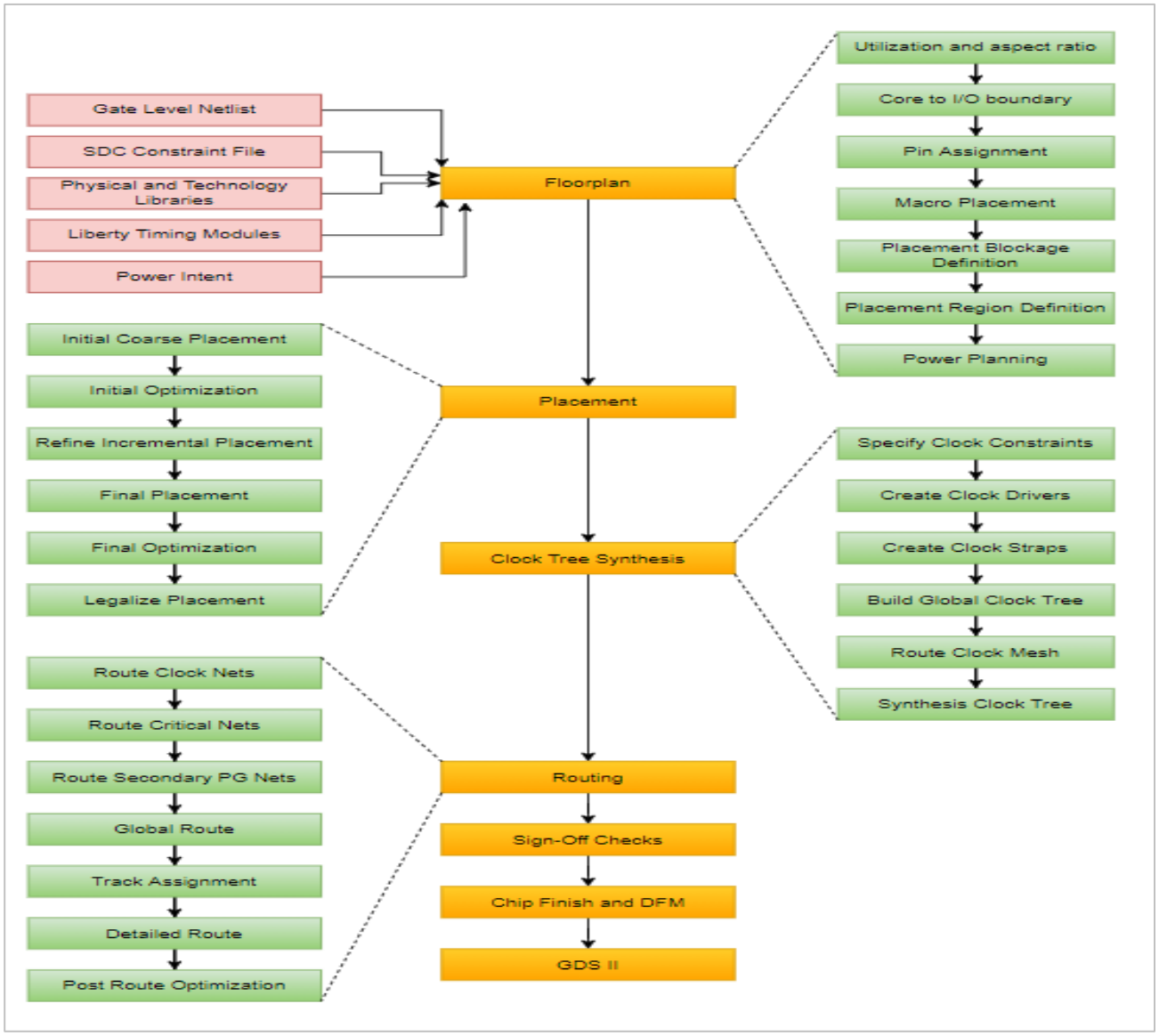

Figure 2 Overall methodology

3.3Floor-plan methodology

In this design, the core utilization is set at 78.54% and the aspect ratio to 0.98. The core to IO boundary is defined for the left, right, bottom, and top edges. The core boundary needs to be decided to snap the boundary points on the instance grid.

3.3.1 Voltage areas

The voltage areas are specified to aid the multi-voltage design as decided in the Unified Power Format (UPF) file. The voltage areas can be nested, disjoint, or overlapping. Guard bands are added to voltage areas to prevent cells from other voltage areas overlapping the present voltage area.

3.3.2 Pin placement

The pins are assigned to different edges of the core based on the top-level module connections as shown in *Figure 3.* The width and pitch for each layer of the pins must be assigned. Even metal layers (M2 and M4) are assigned to vertical pin tracks and odd metal layers (M1 and M3) are assigned to horizontal pin tracks.

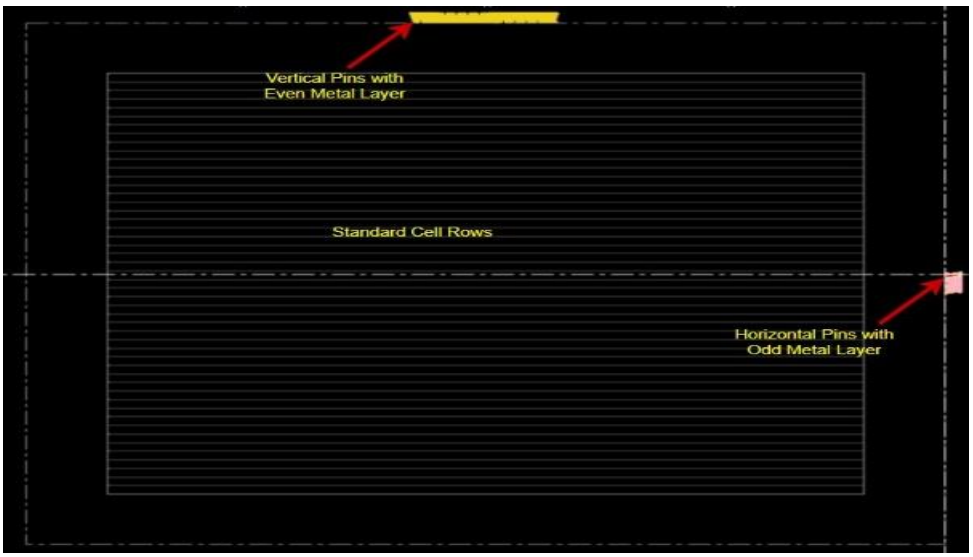

Figure 3 Pin assignment

3.3.3 Macro placement

Macros are generally memories and certain hard Intellectual Properties (IP). The macros have a large number of connections to standard cells as well as the I/O pins. Hence, the placement of these macros is crucial to reduce congestion and wire length. Longer wire lengths lead to an increased transition time, which makes timing closure difficult. Hence, placement of macros based on fly-lines are considered. Fly-lines provide an idea of macro connection to the pins and standard cells. Three macro placement strategies have been explored as under.

1) Island macro placement - All the macros are placed together on one side of the core area forming an island. The island is formed such that a regular rectangular area is made available for standard cell placement as shown in *Figure 4*.

2) Peripheral macro placement - The macros are placed on the periphery of the core area boundary, as close to the pins as possible in *Figure 5*. The core area in the middle is available for the standard cell placement.

3) Donut macro placement - The macros are placed on the periphery as well as in the middle of the core area forming a donut shape to place the standard cells as in *Figure 6*.

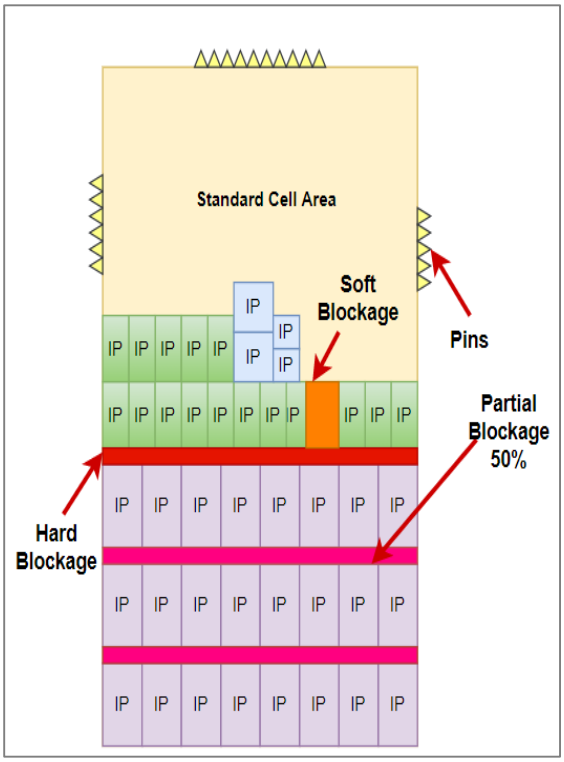

Figure 4 Island macro placement

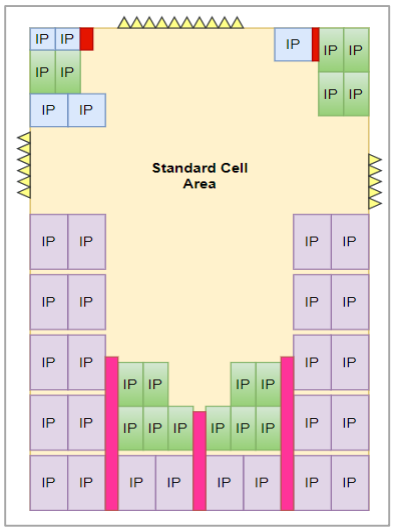

Figure 5 Peripheral macro placement

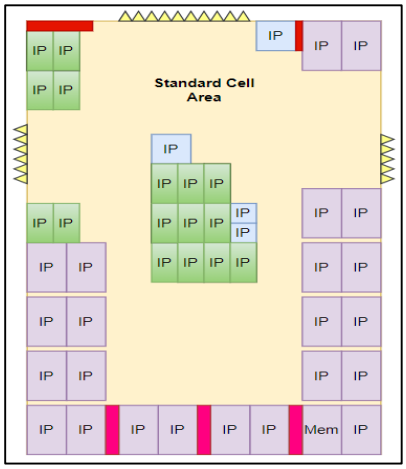

Figure 6 Donut macro placement

3.3.4 Placement blockages

Placement blockages are added to control the placement of standard cells in some regions. The types of placement blockages are:

1) **Hard blockages** - Regions where no standard cells or hard macros can be placed. The regions marked red in *Figure 4* are hard blockages.
2) **Soft blockages** - Regions where no standard cells can be placed during initial placement, however, buffers and inverters can be placed during optimization. The regions marked orange in *Figure 4* are soft blockages.
3) **Partial blockages** - Regions where standard cells can be placed, but only up to a certain cell density as specified. The regions marked pink in *Figure 4* are partial blockages with a cell density of 50%.

3.3.5 Power planning

The power mesh is built in this step to form a power distribution network across the core area. A set of guard rails of Voltage Drain Drain (VDD) and Ground (GND) is built around the core area which connects the primary supply input ports. A set of horizontal and vertical rails of both voltage levels are formed to provide power to all parts of the core as shown in *Figure 7.*

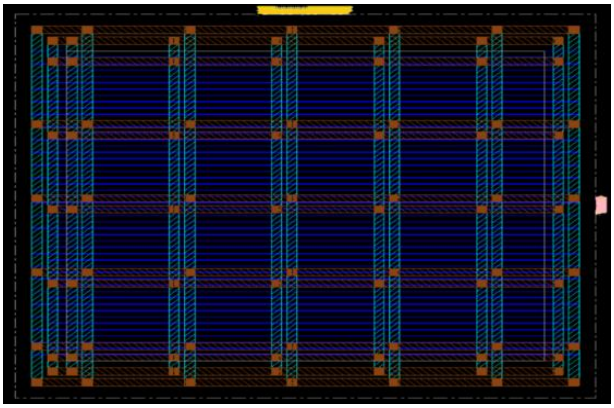

Figure 7 Power stripes

3.3.6 Power switches

Power switches are used in multi-voltage designs. According to the UPF specifications, the power switches are placed in a daisy chain fashion as seen in *Figure 8.*

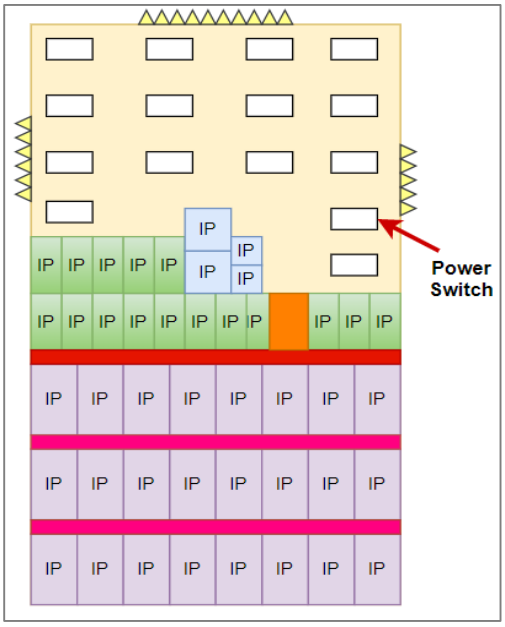

Figure 8 Power switch insertion

3.3.7 Addition of physical-only cells

The final stage of the floor-plan is the addition of the physical-only cells. These cells have no logical functionality. End-Cap cells are added at the edges of the core and around macros to protect the diffusion and poly layers during lithography. Well-Tap cells are added to provide the substrate of the transistors with appropriate well- voltage for proper functionality. Another type of cell added is tie-cell which is used to tie a wire to either logic 1 or 0. These are used to interface between powered down and always-on power domains.

3.4Place and route methodology

3.4.1 Placement preparation

After the floor-plan is completed, the floor-plan specifications are written onto a Design Exchange Format (DEF) file. The inputs to the placement tool are the gate-level net-list, floor-plan DEF file, power intent file, timing module files, and reference library files. Sanity checks are performed on the floor-plan file and gate-level net-list. The power intent file is checked and any violations between the floor-plan data, gate-level net-list, and power intent are corrected. The floor-plan information is then loaded onto the tool. Next, the power intent is committed, which adds the isolation cells, retention cells, enable level shifters, and power Mux-s.

3.4.2 Optimization preparation

Placement optimization is an important step during PD flow. Several parameters and tool options of the Synopsis IC Compiler II tool need to be set before optimization.

1. **Setting target library files** - The library files that should be used by the tool for optimization and clock tree synthesis should be defined by using the *set_target_library_subset* command.
2. **Restricting library cells** - The command *set_lib_cell_purpose* restricts the library cells used during optimization, clock tree synthesis, and setup/hold fixing. This reduces the tool runtime as only specific cells will be tried and tested for optimization.
3. **Preventing optimization on cells** - By setting the *size_only* option on some cells, optimization can be prevented on certain cells. This command is set of cells present in the clock paths.
4. **Setting percentage low Voltage Threshold (VT) optimization** - Low VT cells consume low power but have high leakage current. The *set_max_lvth_percentage* command restricts the use of low VT cells to a defined value and the tool considers leakage and power trade-off during optimization. In this design, the percentage low VT is set to 20.
5. **Specifying routing resources** - The minimum and maximum routing layers globally, and for specific nets can be set. Layers to be ignored for Resistance-Capacitance (RC) estimation during optimization are also set using the *set_ignored_layers* command.
6. **Defining placement bounds** - Placement bounds are of move and group type. Move bounds have a fixed location and boundary, whereas group types have a fixed boundary. The bounds are set to group similar logic level cells to reduce wire length time.
7. **Enable power optimization** - Dynamic power optimization is enabled for the design by using the command *set_scenario_status-dynamic_power true*.
8. **Enabling congestion driven placement** - The congestion effort can be controlled by, *set_app_options -name place_opt.congestion.effort -value high*.
9. **Enable global route estimation** - The optimization engine makes use of a virtual route to estimate wire length for timing fixing. Global routing gives a more accurate estimate of the wire length, but increases the run time. In the design, global routing for placement and high fan-out net synthesis is enabled.
10. **Performing magnet placement** - Magnet placement is used to place certain logic cells close to objects to reduce the wire length. Certain macros are set to act as magnet objects for some logic cells to which they connect extensively.

3.4.3 Performing placement

The *place_opt* command is run to invoke the tool to run placement. Several optimization iterations are performed to get an optimized placement for congestion and timing. The placement is then legalized to snap the standard cells to the manufacturing grid. The placement is checked to resolve any violations before moving to the Clock Tree Synthesis (CTS) stage.

3.4.4 CTS

The CTS starts with deriving the clock trees and checking for all clock constraints. The clock constraints must be specified for all clocks and a clock reference must be derived for all clock cells. The transition and capacitance for each input port must also be specified. The parameters that are set to prepare for CTS are as under:

1. **Enable skew and target latencies** - The tool tries to achieve the skew and target latency values as required by specific designs during the optimization.
2. **Enable local skew optimization and skew groups** - The skew groups are a set of clock cells among

which the skew must be balanced. Local balancing results in a much-optimized timing for clock paths.

3. **Specifying the primary corner** - The optimization tool uses the set primary corner to resolve setup and hold violations. The primary corner defined is generally the extreme corner for which timing must meet the requirements.
4. **Enabling dirty design mode** - The constraints in the SDC file can get extremely tight, which increases the optimization run time. This setting is specified to get optimum results in lesser time as the tool ignores a few constraints to meet timing.
5. **Enabling global route** - Global routing for clock nets is enabled to get accurate wire length timing values during optimization.
6. **Enable Concurrent Clock and Data (CCD) optimization** - The option is enabled to perform optimization on both clock and data paths. Buffers and inverters will be added to the data path to meet and balance timing.

Once the specifications are enabled, a clock tree can be built. The clock tree is first built by inserting the mesh and tap drivers across the core area. The clock mesh is then built using the *create_clock_straps* command. The global clock tree is then built which is generally an H-tree structure. The mesh and tap drivers are routed to the global clock tree and mesh, followed by synthesis and optimization of the entire clock tree. A tap driver connected to the various sinks is shown in *Figure 9.*

3.4.5 Routing

The routing parameters that are set before performing routing area:

1) **Defining routing guides, blockages, and corridors** – Routing guides are regions where specific routing characteristics such as horizontal and vertical track utilization, and preferred routing direction can be fixed. Routing blockages are areas where routing of certain layers is not allowed. Routing blockages are placed close to the pins to reduce routing congestion. Routing corridors are regions where the routing of some nets can be restricted.
2) **Defining Non-Default Routing (NDR) for clock and signal nets** - Certain nets require special route layer characteristics. The trunks of the clock tree are usually routed with a double width layer which is specified as an NDR rule. NDR rules are specified for certain nets after looking at logical connectivity.
3) **Routing clock nets** - The global routing, track assignment, and detail routing is performed for all clock nets.
4) **Routing critical nets** - Certain nets as studied from the data flow logic are considered critical nets. These nets must be routed first to fix them and prevent optimization of these nets further.

Once the clock and critical nets are routed, the routing of the entire design can be carried out. The routing engine first assigns global routes to all nets and overflow in each global route cell is reported. The track assignment is then performed which contains certain violations. The detailed routing routes the nets completely and resolves violations. Post route optimization is performed which includes legalization of cells, incremental detail routing, and ECO routing.

3.5 Implementation specifications

The floor-plan of the LLC module was performed on Cadence Innovus Implementation System and the PNR flow was carried out using Synopsis IC Compiler II. The LLC module specifications are given below in *Table 1.*

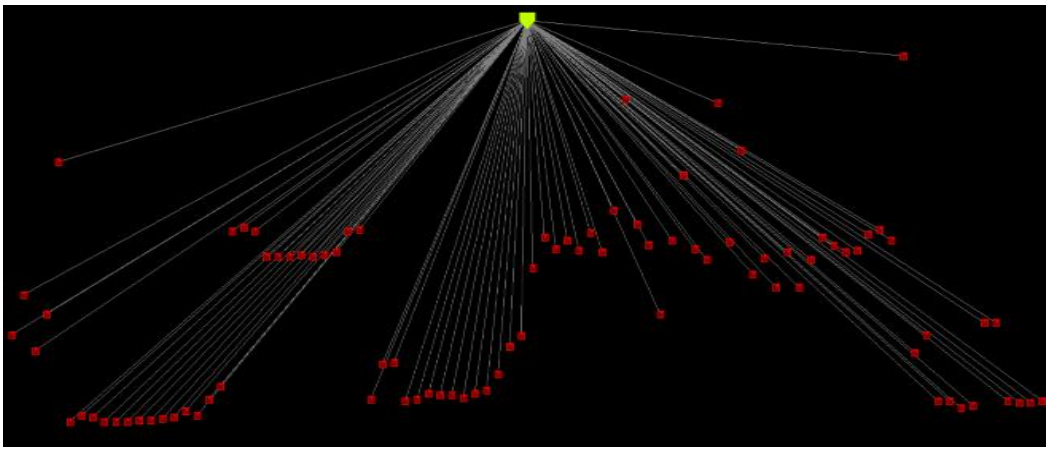

Figure 9 H-clock tree

Table 1 LLC module specifications

Parameter	Number
Inputs	891
Outputs	756
Macros	57
Leaf Instances	125014
Clocks	23
Clock Gating Cells	8675
Registers	97845
Clock Gating Ratio	100
Retention Flops	45689
Buffers	145980

4.Results

The entire PD flow for the three macro placement techniques was carried out using Cadence Innovus Implementation System and Synopsis IC Compiler II. The power, timing, and congestion were monitored at every step. The post-route setup timing is shown in *Figure 10*. The Worst Negative Slack (WNS) is seen to be negative.

```
------------------------------------------------------------
         timeDesign Summary
------------------------------------------------------------

Setup views included:
 slow_max

+--------------------+--------+---------+---------+---------+---------+---------+
|    Setup mode      |  all   | reg2reg | in2reg  | reg2out | in2out  | default |
+--------------------+--------+---------+---------+---------+---------+---------+
|           WNS (ns):| -0.977 | -0.086  | -0.977  | -0.219  | -0.265  |  0.000  |
|           TNS (ns):| -45.777| -0.327  | -44.884 | -0.507  | -0.455  |  0.000  |
|    Violating Paths:|   98   |    0    |   94    |    3    |    2    |    0    |
|          All Paths:|  208   |   168   |   128   |   37    |    5    |    0    |
+--------------------+--------+---------+---------+---------+---------+---------+
```

Figure 10 Post-route setup timing report

The setup timing needs to be optimized such that the WNS is positive. The optimization can be done by upsizing the cells present in the data path, to increase data propagation delay. The optimized setup time report is shown in *Figure 11*, where the WNS is made positive.

```
         timeDesign Summary
---------------------------------------------------------

Setup views included:
 slow_max

+--------------------+--------+---------+---------+---------+---------+---------+
|    Setup mode      |  all   | reg2reg | in2reg  | reg2out | in2out  | default |
+--------------------+--------+---------+---------+---------+---------+---------+
|           WNS (ns):|  0,678 |  0,007  |  0,678  |  0,004  |  0,321  |  0,000  |
|           TNS (ns):| 32,323 |  0,000  | 31,804  |  0,004  |  0,518  |  0,000  |
|    Violating Paths:|   87   |    0    |   85    |    1    |    2    |    0    |
|          All Paths:|  208   |   168   |   128   |   37    |    5    |    0    |
+--------------------+--------+---------+---------+---------+---------+---------+
```

Figure 11 Optimized post-route setup timing report

The placed design of the peripheral macro strategy is shown in *Figure 12*. Before exporting the file to the GDS II format, all the DRC violations were cleared as shown in *Figure 13*.

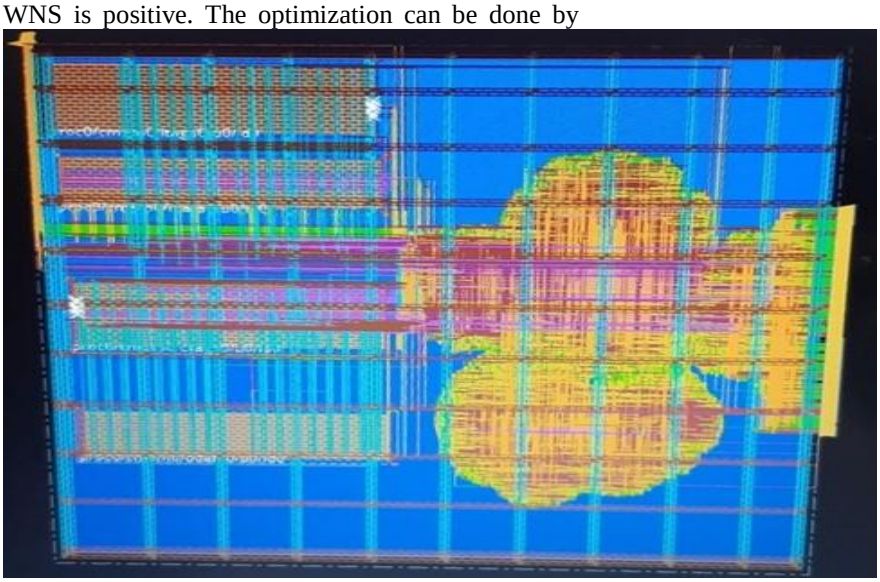

Figure 12 Peripheral macro placed design

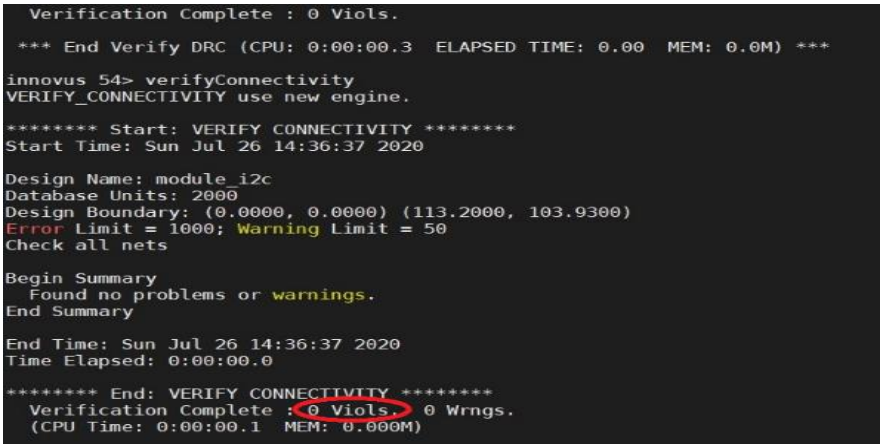

Figure 13 DRC checks

The floor-plan sanity check results are tabulated in *Table 2*. The observations from the sanity check are as follows:

1. The standard cell area for island macro placement is highest, followed by the donut and then peripheral configurations.
2. The blockage area for donut macro placement is the lowest, followed by peripheral macro placement and island macro placement. This shows that the non-uniformity of macro placement is highest for island placement, making it an inefficient macro placement strategy.
3. While the number of cell rows is highest for island macro placement, the number of unique cell rows is the least, which indicates less uniform standard cell placement. In this regard, the peripheral macro placement proves to be the best.
4. The core density and gate density are lowest for peripheral macro placement, moderate for donut placement, and highest for island placement. This proves that the area available for clock and signal routing is more for peripheral placement as compared to the other strategies, thereby reducing congestion of the core area.
5. The number of power switch cells and PG pins placed in the core area is highest for island macro placement, followed by peripheral and then donut. The power switch cells consume extra power and a higher value of these lead to more power consumption of the chip. The peripheral macro placement again is observed to be best with a moderate value for power consumption.
6. The number of Global Cell (GCell) route congestion is a rough indication of the congestion after routing takes place per GCell. The congestion is seen to be minimum for peripheral macro placement, indicating it to be a better macro placement choice among the three.

Table 2 Floor-plan parameter comparison

No	Property	Island placement	Peripheral placement	Donut placement
1	Standard cell area (nm^2)	0.1566	0.1549	0.1550
2	Macro area (nm^2)	0.4047	0.4047	0.4047
3	Blockage area (nm^2)	0.01589	0.00877	0.006320
4	Number of cell rows	27535	22980	22830
5	Gate density	58.83%	56.53%	56.63%
6	Core density	80.30%	80.15%	80.28%
7	Power switch cells	2098	2025	2003
8	Number of core sites	23423786	23493460	23398572
9	Number of unique length rows	14	28	22
10	Number of PG pins	15161	15088	15066
11	Row area (nm^2)	0.2732	0.2740	0.2729
12	Number of GCells with routing track overflow	321	145	382
13	Number of vias	17923	16538	17178

The QOR results at each step of PD implementation flow for each macro placement strategy are given as follows:

1)After standard cell placement- *Table 3* tabulates the QOR after standard cell placement as follows:

i) The overall WNS value is seen to be negative as the design has not been optimized for timing. However, the WNS value is least for peripheral macro placement, indicating better timing QOR.

ii) The power is seen to be minimum for peripheral macro placement and highest for donut macro placement.

iii) The congestion value is moderate for all the strategies as the clock and signal routes have not been placed yet.

2)After CTS- The QOR comparison after CTS is tabulated in *Table 4*. The WNS value has reduced across all the macro placement strategies due to the post-placement optimization that occurs before CTS. The power has increased due to extra power consumption by the clock controllers and cells. The routing congestion is also seen to have increased for each of these as the available density for routing has reduced after the introduction of clock cells.

3)After routing- The WNS value for each macro placement strategy has become less negative, indicating an improvement in timing QOR in *Table 5*. The power is seen to further increase, due to the power consumption of signal and clock rates. The congestion value is increased slightly after CTS. The slight increase is caused due to the routing optimization carried out by the tool.

4)After chip sign-off- The WNS has been optimized to obtain a positive value which indicates better timing closure as seen in *Table 6.* The power is seen to have increased from the routing stage due to the inserted buffers to close timing. The overall routing congestion is seen to be lowest for peripheral placement proving it to be a better macro placement option along with the least WNS and power consumption.

Complete list of abbreviations is shown in *Appendix I.*

Table 3 QOR comparison after standard cell placement

Property	Island placement	Peripheral placement	Donut placement
WNS (ns)	-45.36	-34.8	-49.67
Power (µW)	0.467	0.389	0.521
Horizontal congestion overflow	0.19%	0.12%	0.24%
Vertical congestion overflow	0.23%	0.21%	0.28%
Total congestion overflow (horizontal +vertical)	0.21%	0.165%	0.26%

Table 4 QOR comparison after CTS

Property	Island placement	Peripheral placement	Donut placement
WNS (ns)	-26.89	-23.07	-34.7
Power (µW)	0.551	0.456	0.592
Horizontal congestion overflow	0.38%	0.34%	0.45%
Vertical congestion overflow	0.36%	0.29%	0.38%
Total congestion overflow (horizontal +vertical)	0.37%	0.31%	0.415%

Table 5 QOR comparison after routing

Property	Island placement	Peripheral placement	Donut placement
WNS (ns)	-24.3	-19.6	-31.7
Power (µW)	0.779	0.654	0.967
Horizontal congestion overflow	0.47%	0.32%	0.41%
Vertical congestion overflow	0.35%	0.31%	%
Total congestion overflow (horizontal +vertical)	0.315%	0.315%	0.395%

Table 6 QOR comparison after chip sign-off

Property	Island placement	Peripheral placement	Donut placement
WNS (ns)	0.678	0.57	0.985
Power (µW)	0.9525	0.765	1.043
Horizontal congestion overflow	0.47%	0.24%	0.44%
Vertical congestion overflow	0.26%	0.39%	1.37%
Total congestion overflow (horizontal +vertical)	0.37%	0.32%	0.90%

5.Discussion

The floor-plan stage during the PNR flow is an important step towards designing congestion-free chip layouts. A clean floor-plan considerably improves the QOR parameters for the design, and macro placement plays a big role in clean floor-plan design. The macros have a high number of connections to both the standard cells and IO ports, and an inefficient placement can lead to increased wire lengths. The macro placement must be done keeping in mind the fly-line connections. The peripheral macro placement is seen to be the best providing a 16% improvement in WNS and a 19.6% improvement in power as compared to the island macro placement. The total congestion is also the least for peripheral macro placement, which is 0.32%, making chip finish and metal fill steps after routing easy. On the other hand, the donut macro placement is the worst macro placement strategy with a 45% degradation in WNS and a 9.5% increase in power consumption as compared to the island macro placement strategy.

5.1Limitations

The limitation of this work is that the timing analysis is done only on the LLC module of the DDR subsystem. Once the LLC block integrates as a black box with other sub-blocks of the DDR subsystem, new timing paths might get created which result in negative slack. Hence, the optimization will have to be performed for the LLC block again, keeping in mind the overall timing paths. Another limitation of the work is the limited corners used to optimize the setup and hold timing paths. Only 10 extreme corners were used to perform design closure. The addition of more corners to restrict the design will ensure a more robust design.

6.Conclusion and future work

The increased demand for high-performance electronic gadgets has led to the exploration of the DDR memory subsystem. The timing critical physical interface and complex logical architecture of the DDR need to be handled during PD implementation. An optimized floor-plan leads to reduced congestion of the module; hence, different macro placement techniques were discussed. The complete PNR flow was presented in the paper to optimize timing, power, and congestion for each of the placement strategies. A 16% improvement in timing is observed for the peripheral macro placement and a 19.6% improvement in power is obtained for the peripheral macro placement strategy as compared to the island macro placement. The overall congestion for island macro placement and peripheral macro placement were seen to be 0.37% and 0.32%, respectively, while for the donut macro placement it was 0.9%. The island macro placement and peripheral macro placement can be used to optimize congestion; where island macro placement can be used for high cell density modules, whereas the peripheral macro placement can be used for a lesser cell density module. The peripheral macro placement is proving to be the best placement strategy as compared to an island and donut macro placement strategies.

This work can be extended in the future by including more timing corners to check the design during timing analysis. The inclusion of more corners with different processes, voltage, and temperature variations can lead to a more robust design immune to environmental fluctuations. Moreover, the physical implementation was performed only for the LLC module on the DDR subsystem. The same design methodology can be extended to implement the other blocks present in the design and optimize them for power and congestion.

Acknowledgment

None

Conflicts of interest

The authors have no conflicts of interest to declare.

References

[1] Haseeb K, Din IU, Almogren A, Jan Z, Abbas N, Adnan M. Ddr-esc: a distributed and data reliability model for mobile edge-based sensor-cloud. IEEE Access. 2020; 8:185752-60.

[2] Maity S, Jiang X, Sen S. Theoretical analysis of AM and FM interference robustness of integrating DDR

receiver for human body communication. IEEE Transactions on Biomedical Circuits and Systems. 2019; 13(3):566-78.
[3] Inoue K, Yano Y. A large scale access-control list for IoT security comprising embedded IP-core and DDR DRAM. In international SoC design conference 2016 (pp. 197-8). IEEE.
[4] Hassan M. On the off-chip memory latency of real-time systems: Is DDR dram really the best option? In real-time systems symposium 2018 (pp. 495-505). IEEE.
[5] Behnam P, Bojnordi MN. STFL-DDR: improving the energy-efficiency of memory interface. IEEE Transactions on Computers. 2020; 69(12):1823-34.
[6] Soni A, Soni B, Mehta R. Congestion estimation using various floorplan techniques in 28nm soc design. In international conference on intelligent computing and control systems 2020 (pp. 199-204). IEEE.
[7] Zhang Y, Peng X. A partition level floorplan method based on data flow analysis for physical design of digital IC. In international conference on integrated circuits and microsystems 2017 (pp. 74-7). IEEE.
[8] Garg S, Shukla NK. A study of floorplanning challenges and analysis of macro placement approaches in physical aware synthesis. International Journal of Hybrid Information Technology. 2016; 9(1):279-90.
[9] Chan CK, Wu TM, Wu ML, Fan GJ, Shiah C, Lu NC, et al. Power distribution network modeling and design of re-distribution layer in DDR application. In workshop on signal and power integrity 2020 (pp. 1-4). IEEE.
[10] MP PK, Panda SK. Design and verification of DDR SDRAM memory controller using systemverilog for higher coverage. In international conference on intelligent computing and control systems 2019 (pp. 689-94). IEEE.
[11] Sim SW, Andersson W. On-die decoupling capacitor optimization for DDR IO interface power rail. In conference on electrical performance of electronic packaging and systems 2018 (pp. 229-31). IEEE.
[12] Ejaz A, Papaefstathiou V, Sourdis I. FreewayNoC: a DDR NoC with pipeline bypassing. In international symposium on networks-on-chip 2018 (pp. 1-8). IEEE.
[13] Mohamed J, Michalka T, Ozbayat S, Luevano GR. PDN design and sensitivity analysis using synthesized models in DDR SI/PI co-simulations. In electrical design of advanced packaging and systems symposium 2018 (pp. 1-3). IEEE.
[14] Al-obaidy F, Asad A, Mohammadi F. Power-management based on reconfigurable last-cache level on non-volatile memories in chip-multi processors. In Canadian conference of electrical and computer engineering 2019 (pp. 1-4). IEEE.
[15] Nath A, Kapoor HK. Write variation aware cache partitioning for improved lifetime in non-volatile caches. In international conference on VLSI design and international conference on embedded systems 2019 (pp. 425-30). IEEE.
[16] Sakhare S, Perumkunnil M, Bao TH, Rao S, Kim W, Crotti D, et al. Enablement of STT-MRAM as last level cache for the high performance computing domain at the 5nm node. In international electron devices meeting 2018. IEEE.
[17] Cho H, Kong J, Munir A, Giri NK. CT-cache: compressed tag-driven cache architecture. In computer society annual symposium on VLSI 2018 (pp. 94-9). IEEE.
[18] Jang G, Gaudiot JL. Data shepherding: a last level cache design for large scale chips. In international conference on high performance computing and communications; international conference on smart city; international conference on data science and systems 2019 (pp. 1920-7). IEEE.
[19] Lin JM, Deng YL, Li ST, Yu BH, Chang LY, Peng TW. Regularity-aware routability-driven macro placement methodology for mixed-size circuits with obstacles. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2018; 27(1):57-68.
[20] Lin JM, Deng YL, Yang YC, Chen JJ, Chen YC. A novel macro placement approach based on simulated evolution algorithm. In international conference on computer-aided design 2019 (pp. 1-7). IEEE.
[21] Lin JM, Li ST, Wang YT. Routability-driven mixed-size placement prototyping approach considering design hierarchy and indirect connectivity between macros. In proceedings of the annual design automation conference 2019 (pp. 1-6).
[22] Vidal-obiols A, Cortadella J, Petit J, Galceran-oms M, Martorell F. Multi-level dataflow-driven macro placement guided by RTL structure and analytical methods. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2020.
[23] Uppula V, Kesav SV, Vura B. Impact on the physical design flow, due to repositioning the macros in the floorplan stage of video decoder at lower technologies. International conference on distributed computing, VLSI, electrical circuits and robotics 2019 ((pp. 1-6). IEEE.
[24] Shaikh M, Soni B, Mehta R. Optimization of floorplan strategies to reduce timing violation on 28nm ASIC and scopes of improvement for data center ASICs. In international conference on intelligent computing and control systems 2020 (pp. 93-8). IEEE.
[25] Hu Q, Zhang MS. A collaborative optimization for floorplanning and pin assignment of 3D ICs based on GA-SA algorithm. In international symposium on electromagnetic compatibility & signal/power integrity 2020 (pp. 434-8). IEEE.
[26] Cheng WK, Wu CS. Machine learning techniques for building and evaluation of routability-driven macro placement. In international conference on consumer electronics-Taiwan 2019 (pp. 1-2). IEEE.
[27] Wang L, Wang J, Zhang Q. Design and implementation of DDR SDRAM controller based on FPGA in satellite navigation system. In international conference on signal processing 2012 (pp. 456-60). IEEE.

Juhie Fadnavis is currently pursuing her Bachelors of Engineering from RV College of Engineering, Bangalore. She completed her schooling at National Public School, Bangalore in 2017. Her interests lie in Static Timing Analysis, VLSI Chip Engineering, and Embedded Systems. She wishes to pursue her Master's in Engineering from a reputed University soon.
Email: juhiefadnavis.ec17@rvce.edu.in

Kariyappa B. S obtained his B.E. degree in Electronics and Communication from Bangalore University, in 1997, ME degree in Electronics and Communication from the same university in 2000, and the Ph.D. degree in Electronics and Communication from Avinashlingam University, Coimbatore in 2012. He is currently working as Professor in the Electronics and Communication Engineering department of R V College of Engineering, Bengaluru. With over 20 years of teaching experience, he is guiding 3 Ph.D. students and guided many undergraduate and postgraduate student projects. He has authored/co-authored more than 60 articles in refereed international journals/conferences and having a good number of Scopus and Google scholar citations.
Email: kariyappabs@rvce.edu.in

Appendix I

S.No.	Abbreviation	Description
1	CAD	Computer-Aided Design
2	CCD	Concurrent Clock and Data
3	CTS	Clock Tree Synthesis
4	DDR	Double Data Rate
5	DRAM	Dynamic Random-Access Memory
6	DRC	Design Rule Check
7	DSP	Digital Signal Processor
8	DEF	Design Exchange Format
9	ECO	Engineering Change Order
10	GCell	Global Cell
11	GDS	Graphic Data Stream
12	GSA	Genetic and Simulated Annealing
13	GPU	Graphics Processing Unit
14	GND	Ground
15	I/O	Input/Output
16	IoT	Internet of Things
17	IP	Intellectual Properties
18	LLC	Last Level Cache
19	MRAM	Magnetic Random Access Memory
20	NDR	Non-Default Routing
21	NoC	Network On-Chip
22	PC	Personal Computers
23	PD	Physical Design
24	PDN	Power Delivery Network
25	PNR	Place and Route
26	PG	Power/Ground
27	QOR	Quality of Results
28	RC	Resistance-Capacitance
29	RDL	Re-Distribution Layer
30	RLC	Resistance-Inductance-Capacitance
31	RTL	Register Transfer Logic
32	SDC	Synopsis Design Constraints
33	SRAM	Static Random-Access Memory
34	SoC	System-on-Chip
35	STT	Spin Transfer Torque
36	UPF	Unified Power Format
37	VDD	Voltage Drain Drain
38	VLSI	Very Large-Scale Integration
39	VT	Voltage Threshold
40	WNS	Worst Negative Slack