Journal of
Low Power Electronics and Applications

MDPI

Article

A New Physical Design Flow for a Selective State Retention Based Approach

Joseph Rabinowicz [1] and Shlomo Greenberg [1,2,*]

[1] Department of Electrical and Computer Engineering, Ben Gurion University, Beer-Sheva 84105, Israel; rabinowi@post.bgu.ac.il
[2] Department of Electrical Engineering, Sami Shamoon College of Engineering, Beer-Sheva 84100, Israel
* Correspondence: shlomo.greenberg@gmail.com or shlomog@bgu.ac.il or shlomgr@sce.ac.il

Abstract: This research presents a novel approach for physical design implementation aimed for a System on Chip (SoC) based on Selective State Retention techniques. Leakage current has become a dominant factor in Very Large Scale Integration (VLSI) design. Power Gating (PG) techniques were first developed to mitigate these leakage currents, but they result in longer SoC wake-up periods due to loss of state. The common State Retention Power Gating (SRPG) approach was developed to overcome the PG technique's loss of state drawback. However, SRPG resulted in a costly expense of die area overhead due to the additional state retention logic required to keep the design state when power is gated. Moreover, the physical design implementation of SRPG presents additional wiring due to the extra power supply network and power-gating controls for the state retention logic. This results in increased implementation complexity for the physical design tools, and therefore increases runtime and limits the ability to handle large designs. Recently published works on Selective State Retention Power Gating (SSRPG) techniques allow reducing the total amount of retention logic and their leakage currents. Although the SSRPG approach mitigates the overhead area and power limitations of the conventional SRPG technique, still both SRPG and SSRPG approaches require a similar extra power grid network for the retention cells, and the effect of the selective approach on the complexity of the physical design has not been yet investigated. Therefore, this paper introduces further analysis of the physical design flow for the SSRPG design, which is required for optimal cell placement and power grid allocation. This significantly increases the potential routing area, which directly improves the convergence time of the Place and Route tools.

Keywords: physical design; power grid; power-gating; SRPG; selective SRPG; floorplanning; place and route

check for updates

Citation: Rabinowicz, J.; Greenberg, S. A New Physical Design Flow for a Selective State Retention Based Approach. *J. Low Power Electron. Appl.* **2021**, *11*, 35. https://doi.org/10.3390/jlpea11030035

Academic Editor: Alex Serb

Received: 28 July 2021
Accepted: 9 September 2021
Published: 13 September 2021

1. Introduction

Leakage currents during standby mode become more significant in mobile devices as semiconductor processes continue to shrink [1]. These static leakage currents impact the battery standby time of low-power mobile devices when they are in an idle state. Therefore, to mitigate the static leakage currents, some Power-Gating (PG) techniques were developed [2–6]. Power-gating eliminates the static leakage but with no intention to retain the system state. As mobile devices are required to support many features and functions, resulting in a wide range of multitasking, a minimum delay for the state restoration of all active tasks is critical for user satisfaction [7]. Besides the additional delay, saving and restoring the system state presents additional dynamic power overhead that may not be acceptable for certain common applications.

Scan-based techniques, which are used for serially saving and restoring internal retention cells, also suffer from latency and energy overhead [8]. The State Retention Power Gating (SPRG) technique addresses the above-mentioned PG technique's limitations [9–13]. This technique uses unique retention cells to retain the flip-flops (FFs) values during power down (standby state). These cells have been widely adopted in standard library cells of major FAB vendors (such as TSMC). The SRPG approach aims to retain the systems state during standby, thus eliminating the disadvantages of the power-gating technique. However, common SRPG implementations require additional retention cells for all the FFs in the design resulting in significant area overhead. Moreover, these retention cells need to be connected to a dedicated power supply network and retention control signals. This additional wiring increases the area overhead and also complicates the physical design implementation in terms of tools runtime and the ability to handle large designs.

A more advanced approach, called Selective State Retention Power Gating (SSRPG), dramatically reduces the SRPG area overhead and further decreases the static power consumption. The main idea is to find a minimized set of FFs which are sufficient to retain the system state during standby. Chiang et al. [14] propose an empirical nonformal method for the selection of registers whose retention is unnecessary. Darbari et al. [15] present a formal approach based on symbolic simulation for implementing selective state retention. However, this method requires a formal representation of the entire design, which is not always available, and also no automated techniques are proposed. The two recently published SSRPG approaches introduced by [16,17] provide pure formal methods for automatic selecting of all the FF's, which require retention and are essential for a proper system recovery upon power-up. Experimental results show a significant reduction of about 80% of the retention cells area overhead. Recent SSRPG techniques can be efficiently applied to new modern SoC designs for automatic selection and formal validation of essential FFs requires retention. The current work is based on our previous formal SSRPG approach presented in [17], which utilizes formal verification methods and therefore can be easily implemented using the new proposed physical design flow.

Although the SSRPG approach mitigates the area and power overhead limitations of the conventional SRPG technique, still both SRPG and SSRPG approaches require a similar extra power supply network for the retention cells. The impact of the extra power supply when applying the selective approach has not yet been investigated. Therefore, further analysis of the physical design flow for SSRPG design is needed for optimal cell placement and power grid allocation. This may significantly increase the routing area, which in turn directly improves the convergence time of the place and route tools [18].

Furthermore, minimizing the number of retention FFs not only results in reducing the area overhead but also reduces the additional wiring required in SRPG. Although it is shown in [16] that a significant potential area reduction of about 9% of the chip area can be achieved, the added wiring required in SRPG is ignored. In SSRPG, the retention cells footprint can be simply deducted from the total cell area, but the wire-length deduction is not straightforward since it can only be obtained after completing the physical design flow. The wire-length overhead in the SRPG approach is derived from: (1) The connectivity of the retention cells to a new non-gated power supply network [19], and (2) the addition of retention control signals, which need to be connected to all FFs that are being preserved during standby by using retention cells [20]. This wiring overhead complicates the place and route physical design stages in SRPG. This work demonstrates the benefit of applying the SSRPG approach in a real physical design implementation concerning area, power saving, back-end runtime, and wire length.

Although some previous research works [16,17] try to estimate the area and power-saving factor results from applying the SRPG selective approaches, none of them validate it on real physical design implementation. Hence, one of the main objectives of this work is to quantify the real area and power saving factors while using SSRPG comparing to SRPG.

This work also demonstrates the benefit of applying a new, improved localized physical design flow using unique placement rules. The proposed localized improved flow yields significant power supply network area reduction in cases where selective state retention is used. It is shown that by applying these placement rules, metal layers that were originally used for power-supply distribution are freed up to be used for signal routing applied when connecting the different logic gates in the physical design during the routing stage, and therefore improving the routeability. This simplifies the implementation of selective state retention in the physical design flow and significantly reduces the tools' runtime.

Although the SSRPG approach [16,17] is not a new technique, the effect of the selective approach on the complexity of the physical design has not been yet investigated. Therefore, further analysis of the physical design flow for SSRPG design is needed for optimal cell placement and power grid allocation. This may significantly increase the potential routing area, which in turn directly improves the convergence time of the Place and Route tools. This paper aims at the physical implementation aspect to facilitate the complexity of the physical design suggesting a unique flow to efficiently address SoC design based on SSRPG. Moreover, this is the first work related to SSRPG implementation, which accurately quantifies the area, power, and tool runtime saving factors.

In this work, we provide a case study showing the accurate area, power, and tool runtime savings when comparing the physical design implementation of SSRPG to SRPG. Previous works provide area reduction estimations based on the percentage of FFs that does not require retention [16,17]. These area estimations suffer from inaccuracies since they do not take into account the additional wiring overhead required for connecting the retention cells to the non-gated power supply and power-gating controls. To quantify the selective state retention physical design flow benefits, a complete CMOS 28 nm physical design flow was carried out on a typical Double Data Rate (DDR) memory interface controller design.

This paper is organized as follows: Section 2 provides an improved physical design flow for an Application-Specific Integrated Circuit (ASIC) supporting state-retention. Section 3 describes the experiment and shows the comparison results for the four different physical design flows: no retention, full retention using SRPG, SSRPG without special placement rules, and an improved physical design flow for SSRPG. Finally, Section 4 summarizes the paper and states the conclusions.

2. An Improved SSRPG Physical Design Flow

We propose a new approach to the common SRPG technique, based on automatic classification of each of the design's FFs into one of two types: essential or non-essential. The flow begins with gathering the libraries and floor planning, followed by place and routing, and ends with verification of the physical design. Figure 1 depicts the five main stages of a typical physical design flow. Each stage is described in detail in the following section considering the specific additional requirements for state-retention. Two different physical SSRPG design flows are considered concerning the placement stage: distributed flow and improved localized SSRPG flow. Some unique placement rules are proposed for the implementation of the new localized SSRPG physical design approach.

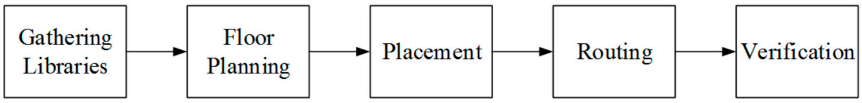

Figure 1. High-level stages of the physical design flow.

Although some physical implementation steps can be controlled by the common UPF and CPF industrial tools for power-aware content, those tools do not provide any specific placing rules except for limiting the logic cells placement to the appropriate power-domain (PDN).

2.1. Gathering Libraries

The libraries' physical design flow contains the list of basic cells and their attributes, such as physical layout abstractions, timing delay models, functional models, and transistor-level circuit descriptions [21].

To implement state retention, the libraries should contain special retention FFs. Such FFs are divided into different types that can be categorized by the two following criteria: (1) the transistors threshold voltages (low, high, or multi-threshold) (2) Using an additional latch (referred to as balloon latch) or rather than using the FF slave latch (in a common master-slave FF) for retention. Table 1 depicts the different types of retention FFs that are used in state retention approaches and their impact on low power, propagation delay, and physical design flow [13,22,23].

Table 1. Retention FFs types and tradeoffs.

FF Type	V_{TH}	Extra Latch	Low Power	Propagation Delay Impact	SoC Physical Design Flow Impact
1	Low	No	Week	Negligible	Need clock and reset gating during standby.
2	Low	Yes	Medium	Negligible	Additional area impact of the balloon latch.
3	High	No	Good	High	Need clock and reset gating during standby.
4	Multi	Yes	Good	Negligible	Extra balloon latch and extra power supply.

Retention FF's implemented with low threshold voltage transistors have less impact on the propagation delay since the low voltage threshold allows fast switching between off and on states. However, since the leakage increases exponentially when decreasing the threshold voltage, the efficiency of reducing the static leakage is limited for this type of FF. The static leakage is given by the following equation:

$$P_{leakage} = V_{dd} \cdot I_{leakage} = V_{dd} \cdot I_0 \cdot \exp\left\{ [(V_{GS} - V_{TH})/V_T]/[1 - \exp\left(\frac{-V_{DS}}{V_T}\right)] \right\} \quad (1)$$

where V_{TH} is the threshold voltage of the transistor, V_T is the thermal voltage, V_{GS} is the voltage between gate and source, and V_{DS} is the voltage between drain and source of a MOSFET transistor. Some improvement in static leakage reduction can be achieved by adding a specific balloon latch, as shown in Figure 2. This additional latch is designed to consume less power during standby since it does not affect the master-slave functional path and therefore supports higher frequencies compared to FFs that use the slave latch for retention.

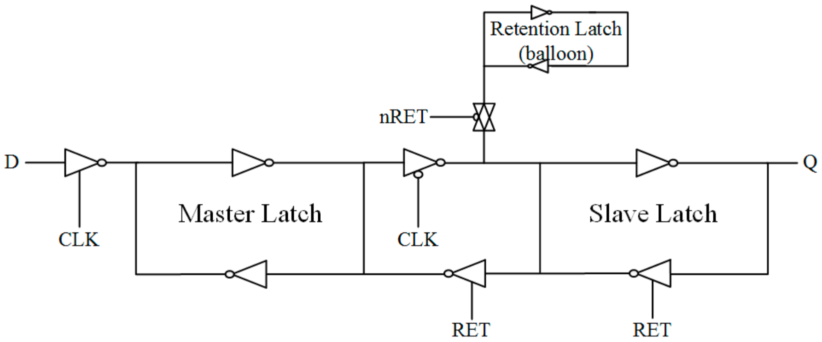

Figure 2. Retention FF implementation using a balloon latch.

Retention FFs that are implemented with high threshold voltage transistors, perform better with respect to static leakage reduction. A high voltage threshold leads to a better closure of the source/drain channels and thus preventing leakage currents when the transistor is in its off state. However, a high voltage threshold also impacts the propagation delay and therefore limits the clock frequency rates. Using both multi-voltage threshold transistors and an additional retention balloon latch allows better static leakage reduction and higher clock frequencies. However, this is at the expense of additional area overhead and extra external SoC power supply, which requires dedicated supply pads and balls, complicating the design [22]. Therefore, while choosing the physical design libraries in case of state retention, the SoC designer should consider the following factors and their tradeoffs: clock frequency, static leakage reduction, area overhead, and implementation complexity.

2.2. Floorplanning

A well-thought-out floor plan leads to a design with higher performance and optimum area [21]. In this stage, the physical designer determines the size of the macro instance, which includes the physical representation of the design. Additionally, the structure and placement of the power and ground strips referred to as power-supply networks are determined.

Some industrial SoCs may contain several power-gated domains and, therefore, many power switches to reduce IR drop [24]. This work aimed specifically at low power designs and referred to the hard macro level of implementation using only one or two power switches (as illustrated in Figure 3). To maintain minimum voltage drop and to prevent performance degradation, the power and ground strips should be as dense as possible. The following section refers to specific floorplanning adjustments required for state-retention-based designs. State-retention approaches require some modifications to the typical floorplan with respect to the power supply network. Specifically, two kinds of floorplan modifications are required: (1) adding an extra retention power supply network and (2) integration of dedicated sleep transistors for disconnecting the main power supply on standby. Figure 3 illustrates two power grids networks with a single power switch. The extra power grid uses a significant portion of the metal layers, which are actually needed for routing the logic gate connections (routeability) [13]. Although the strips of the extra power supply network are thinner compared to those of the main power supply, since there is no need to support full clock rate in standby, they should be spread over the entire macro instance.

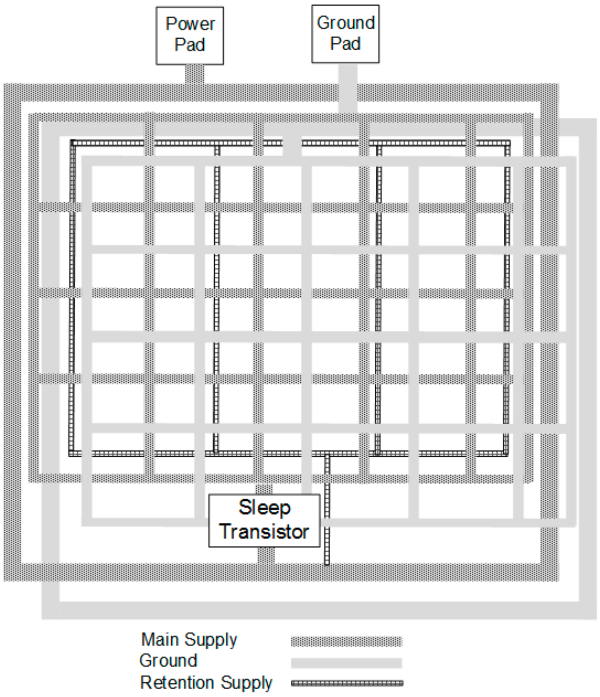

Figure 3. Power grids networks for State Retention-based SoC.

Any power gating implementation, including SRPG, requires a dedicated sleep transistor per gated power supply. The sleep transistors are based on high voltage threshold transistors and are responsible for disconnecting both the power supply source and the ground in standby, as shown in Figure 4. Unique SLEEP signals are used to control the sleep transistors and define two control modes: active and standby modes (SLEEP is driven to 1 during standby and 0 during active modes). The active mode utilizes the low voltage threshold transistors to operate at higher frequencies. In Standby mode, the SLEEP signals are activated to turn off the sleep transistors. Since the sleep transistors are based on high voltage threshold transistors, their static leakage is very small during standby. The size of the sleep transistor is critical in terms of performance, area, and leakage current [19]. While the sleep transistor should be large enough to drive sufficient current to meet frequency performance, it should not cause excessive leakage.

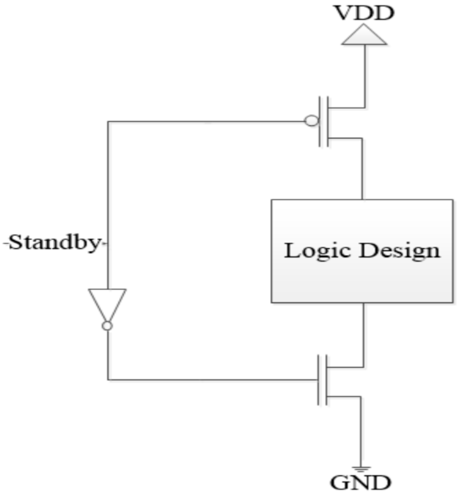

Figure 4. Sleep transistors.

2.3. *Place and Route*

The placement stage is responsible for placing the overall standard logic gates in a given macro instance and inserting buffer cells along with the clock and reset signal paths. Since the long wiring induces different propagation delays between different FFs, a clock balancing process is required. The buffer cells are used both for clock balancing and to support high fan-out and long wiring. This process of buffer insertion is commonly referred to as Clock Tree Synthesis (CTS) and has a significant impact on timing closure. In addition to the clock and reset signals, the CTS process is also applied to the retention FFs' control signals. This wiring and buffering overhead to support the additional retention control signals is significant in designs that include many sequential elements and might be similar to the overhead of the clock network [20]. Since the additional buffers should be connected to the retention power supply network, they have a significant impact on the routing to support the distributed retention controls signal paths. Power-supply network optimization is usually carried out after placement and before signal routing. The objective is to reserve more chip area for signal routing and, at the same time, maintain the performance of the power supply network. However, it is difficult to fully utilize the reserved chip-routing resource [25], especially in the case of a design that requires a dedicated power supply for the retention cells. Therefore, minimizing the area of the retention power supply network will lead a better routing utilization. The routeability in an SSRPG design can be further improved due to the small number of the required retention cells compared to SRPG. The routeability improvement can be achieved by making some appropriate adjustments both in the floorplan and the placement stages.

This work considers two different flows for SSRPG: the more straightforward distributed flow and a unique localized flow. In the distributed flow, the retention FFs are distributed all over the hard macro, while in the localized flow, the retention FFs are placed in a limited area using some placement constraints. Therefore, the region of the PDN of the always-on domain becomes smaller and requires less routing overhead. Furthermore, the proposed physical design flow is implemented within a hard macro level and applied to a specific functional design module. Therefore, since each hard macro commonly contains only one or two power domains, it is feasible to place all the retention FFs, connected to the always-on domain of the specific PDN, within a localized concentrated area.

We propose a unique physical design approach that is based on the assumption that the retention cells can be placed all together in a localized and relatively small area within the entire macro instance. This will lead to a reduced retention power supply network area. Figure 5 depicts placement results for two different physical design flows carried out on the proposed DDR controller design using the Cadence Encounter tool. Figure 5a shows the placement results for the distributed SSRPG flow in which the retention power grid (i.e., power supply network) is distributed throughout the entire macro instance area without any placement constraints as in the common SRPG flow. The figure depicts the spreading of the retention FFs. Figure 5b shows the placement results for the new proposed localized flow. It can be noticed that the retention FFs are now located together in a relatively small localized area.

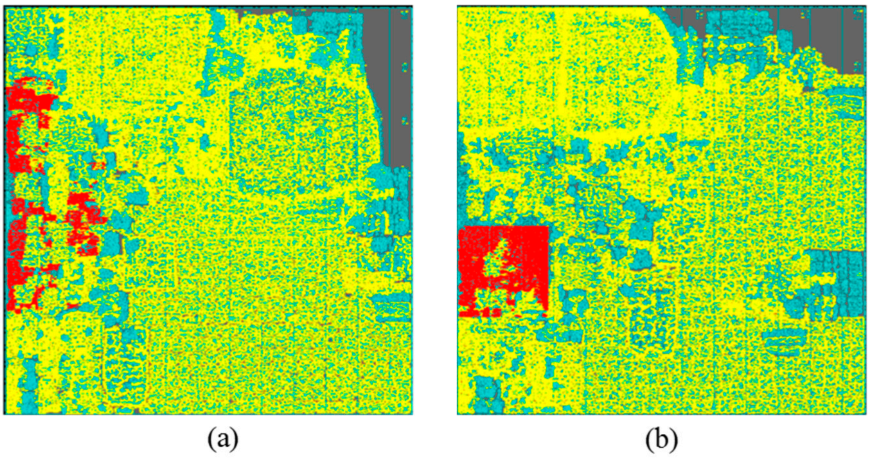

Figure 5. (**a**) Placement of the distributed retention FFs (mark in red and spread mostly on the mid-left-hand side). (**b**) Placement for the proposed flow where retention FFs are placed in a localized area (red square on the mid-left-hand side).

Two modifications were applied to the localized physical design flow based on the distributed flow placement results and using the common SRPG flow. First, the power grid was limited to a specific and localized area in the floorplan stage. Then, some specific placement constraints were provided to the Encounter tool, forcing all retention cells to be placed in a limited minimized localized area within the retention power grid region. The results show that the retention cells and the relevant retention power grid were successfully placed in a minimized area enabling better routeability compared to the common approach. Since the extra power grid utilizes only a small part (about 1/16) of the metal layer used for the retention power supply network (Figure 5b), more metal area is freed up for routing. To further reduce wire-length and additional buffers, the external retention control input ports are also placed in the same selected area close to the retention power grid. Applying such constraints to the placement tool may result in timing violations since the interconnect length between FFs may significantly increase. However, since the number of retention cells in SSRPG is relatively small, and most of the retention FFs are not part of the data path, the timing violations are not critical [26]. In the next stage, the routing process is carried out. Routing is becoming more difficult, especially for state retention-based designs, like SRPG, since the design is getting more complex due to the additional retention cells and the required extra wiring. Therefore, SSRPG facilitates the routing process by significantly reducing the amount of routing and hence decreasing the route runtime.

2.4. *Verification*

The final stage of any physical design flow is verification. This stage focuses on functional testing and design manufacturability. A comprehensive design verification process consists of three categories: functional, timing, and physical. The functional verification includes logic simulations, formality checks, simulation randomization, in-circuit emulation, and hardware/software co-verification [27]. The timing closure is carried out using Static Timing Analysis (STA) to verify the timing of a digital design [28]. The physical verification checks the design layout against the specific process rules and includes Layout Versus Schematic (LVS) and Design Rule Check (DRC) [21]. In the case of state retention, some additional logic simulations scenarios should be considered. For example, entering standby and then restoring the design state upon power resumption and verifying the selection of the appropriate FF's which required retention.

3. Experiment and Results

In this section, we compare four different approaches in respect to the physical design flow: no retention, full retention using SRPG, SSRPG with no specific placement rules, and an improved SSRPG flow. All the flows were applied to a typical DDR controller design as a test case. The synthesis was carried out using the Cadence RTL compiler, and then a common full PD flow was applied using Cadence Encounter to each of the four approaches. One of the main purposes of this work was to quantify the efficiency of the selective approaches with respect to area and power saving. Additionally, this research compares the four different PD flows in respect to the ability of the tools to converge, tools runtime, total wiring length, static leakage, and area-saving factors. Figure 6 depicts the block diagram of the selected DDR controller design. The DDR controller contains about 62,000 FFs. The design contains a DDR control unit, a DDR PHY adaptor, and two ARM AXI bus interfaces. The control unit is used to configure the DDR controller and monitor the status registers. The DDR PHY interface is connected directly to the DDR PHY, while the AXI bus interfaces between the DDR PHY adaptor and the internal memories. The AXI bus is used to store and retrieve data to/from the internal memory using a First-in-First-out (FIFO) memory within the AXI interface. A clock generator is used to provide an accurate clock signal to the external DDR memory. The DDR controller has two different operating modes: consecutive and interleaving memory addressing. The DDR interleave mux selects the desired operating mode and supports data interleaving from two channels to one memory device, reducing the external memory access time. The chosen DDR controller is used in many common VLSI applications and is large enough to represent a typical macro instance. Moreover, the design has a significant amount of non-essential FFs and, therefore, can be efficiently implemented using the SSRPG flow. In addition, the working frequency of the DDR controller is relatively high (533 MHz) and makes the comparison qualify for high-frequency designs as well.

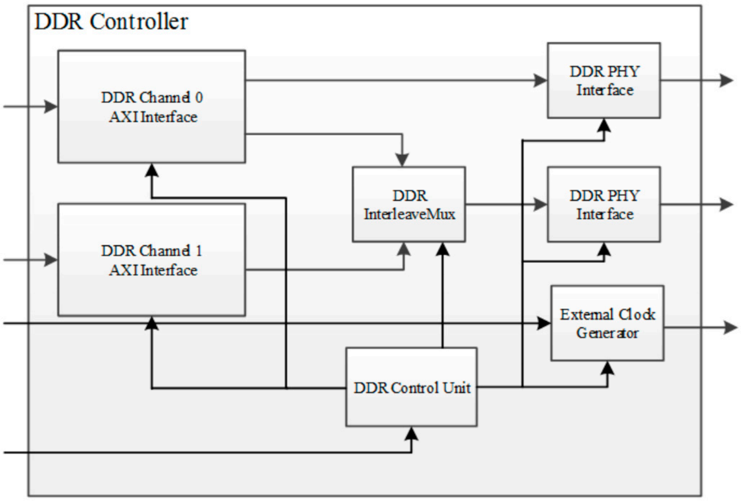

Figure 6. DDR Controller—block diagram.

3.1. Basic Synthesis

Physical Design Flow Implementation

The design was first synthesized using the Cadence RTL compiler (RC). The synthesis results provide the physical designer with the following data: (1) a standard library cell design representation referred to as netlist, (2) the total cell area estimation needed for floorplanning, and (3) critical timing paths that should be addressed in the synthesis stage. For timing closure, the clock frequencies and some specific timing constraints should be defined in the synthesis stage. In our test case, two frequencies were applied: 533 MHz for the AXI bus and DDR PHY interfaces and a lower frequency of 133 MHz for the control logic.

The delay constraints take into consideration 30% of the clock period for output ports and 70% for input ports. Some more delay adjustments were needed for certain ports according to specific timing issues. In order to extract the essential FFs for the DDR controller test case, we have used the SSRPG approach described in [16]. This approach is based on a gate-level analysis and suggests a fully automatic algorithm to classify the FFs in a typical design into two categories essential and non-essential FFs. Results show that only 2522 FFs (out of the total 61,944 FFs) were classified as essential FFs, and therefore only 4.1% of the FFs require retention cells. The netlist was updated accordingly with the additional retention cells.

3.2. Floorplanning

An important step in floor planning is to specify the appropriate area to place macros and standard cells. In general, the floorplan can be determined according to the dimensions of the total macro area, Utilization Factor (UF), and die area. The utilization factor is defined as follows [29].

$$\text{Utilization Factor} = \frac{Area\ of\ Standard\ cells}{Total\ Physical\ Design\ Area} \tag{2}$$

This means that a larger area of 1/UF multiplied by the standard cell area is allocated for the Encounter tool to place the standard cells and to permit enough routing resources for the cells' interconnections. Selection of the UF should both provide the Encounter tool with enough space to place the cells and route between them and still meet timing. As the UF decreases, the area to place cells increases, and therefore the Encounter tool has a better ability to successfully route the cells. The effects of choosing a Utilization Factor on total wire length, congestion, and DRC (Design Rule Constraints) violations have been explored (studied) in [21]. It was observed that a Utilization Factor of 0.5 to 0.7 is appropriate depending on the metal layers in which the Power and Ground planning is done.

The Cadence Encounter tool was used to determine the size of the macro instance for the chosen DDR Controller design. The total cell area (including FFs and logic gates) was extracted from the synthesis results for the four different physical designs. The utilization factor's selection should be considered a tradeoff between the motivation to minimize the macro instance area and the need to reduce the place and route complexity.

An initial recommended utilization factor of 0.7 was examined in the floor planning stage. Then a unique utilization factor was chosen for each of the four different proposed physical design flows according to congestion and DRC violations which directly affect the Encounter tool runtime.

For the no-retention physical design flow, the initial recommended utilization factor of 0.7 was found to be appropriate and did not have much effect on congestion, placement run time, and tool convergence compared to lower utilization factors. However, while applying this initial utilization factor for the SRPG and SSRPG physical design flows, the runtime was significantly higher (a factor of 5) compared to lower utilization factors.

Figure 7 shows the empiric place and route tool's runtime versus the utilization factor for various examined flows. The utilization factor (UF) is given in Equation (2). The available area for placing the cells increases as the UF factor decreases, and therefore the Encounter tool has a better ability to successfully route the cells.

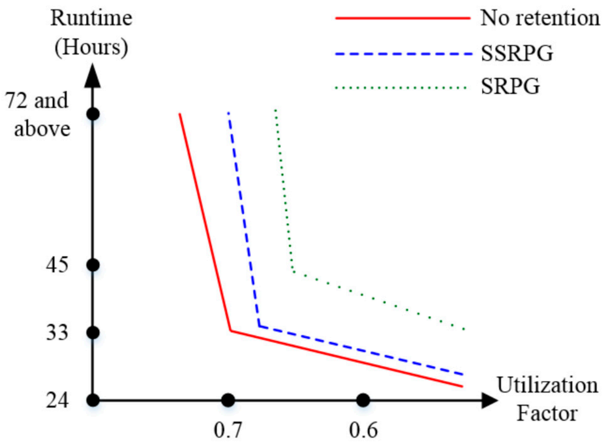

Figure 7. Place and Route runtime versus UF.

The effects of choosing a utilization factor on total wire length, congestion, and DRC (Design Rule Constraints) violations have been explored in [21]. The authors show that by using fewer number of metals to route between the standard cells spread across the core area (which is equivalent to the scenario of less available routing area), the tool has to do complex de-tour routing to avoid DRC violations. It was also observed that with fewer metals (a higher UF), the tool has fewer routing tracks to route between all the cells, introducing more congestion. Therefore, the number of available routing tracks available also decreases.

From Figure 7, we observe that the optimal UF factors are: 0.7, 0.65, and 0.67 for the no-retention, SRPG, and both SSRPG flows accordingly. Any attempt to increase those chosen utilization factors resulted in the divergence of the Encounter tool. In all our experiments, the convergence time limit was defined to be 72 h. The relatively lower UF factor achieved for the SRPG and SSRPG can be explained due to the additional extra power grid and its connections to the retention cells buffers required for the CTS process and the additional route connectivity. We observed that the UF for the SSRPG flow is higher than the UF obtained in the case of SRPG. This means that the SSRPG physical implementation required less area compared to SRPG.

As a part of the floor planning, certain physical elements, such as antenna and latch-up cells, were added to maintain the integrity of the macro instance [30]. Then, pin placement was done according to the SoC constraints. Finally, the appropriate power grid was defined according to the specific physical design flow. While in the case of no-retention flow, only one power grid is required and is spread out uniformly across the macro instance area, the SRPG and SSRPG flow require an extra power grid which should be connected to the additional retention cells.

Figure 8 shows a snapshot, taken from the floorplanning tool, of the two power grids required in SRPG and SSRPG. The common VDD grid is represented by the thick purple line wrapped by two thin red lines. The extra VDDG power grid is represented by two closely placed thin red lines. Since the VDDG supplies power only to the retention cells, it can be composed of fewer gridlines compared to VDD. It can be observed that the VDDG strips are less dense and are placed in a 1.8 µm interval once every second VDD strip. The distance between the VDD and VDDG grid lines was set to 0.125µm. These power grid configurations were validated using the Cadence encounter power analysis tool.

As discussed in Section 2.3, the power grid distribution in the localized SSRPG flow can be limited to a localized area in the floorplan. The exact flow used to determine the localized area in which the retention cells are located is described as follows. First, the floorplan with a uniform distributed power grid is used as an input to the placement stage. Then the results of this placement (location of the retention cells) are used to create a new floorplan in which the power grid is limited to a specific area. Finally, the retention control signals (RETN) which should be connected to all the retention cells, are placed close to this specific region to reduce routing.

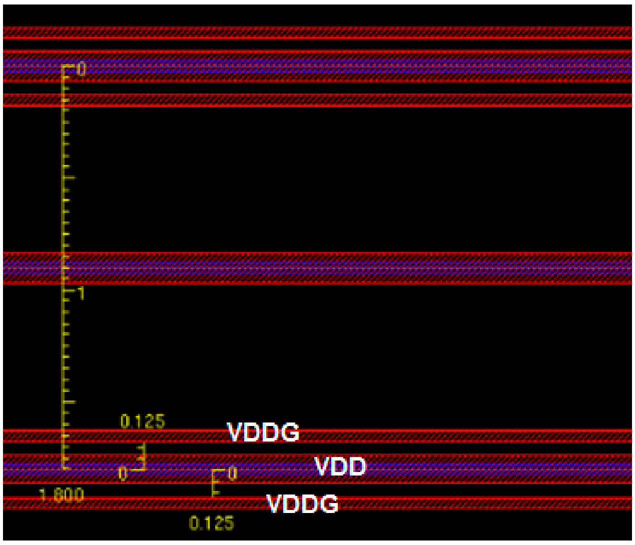

Figure 8. VDD and VDDG power grid floorplanning.

3.3. Placement and Routing

The placement stage was carried out the same way for the four physical design flows. The Cadence Encounter was used as the placement tool in order to meet timing and area constraints as derived from the floorplanning stage. The same clock tree methodology was used for the four examined flows using the CTS Cadence tool with the same timing constraints. In the case of SRPG and SSRPG flows, the additional RETN control signals used for retention purposes were also balanced in the clock tree process. The routing for the three-state retention flows also included the additional connections of the state-retention cells to the extra VDDG power grid.

3.4. Results

During the implementation of the four physical design flows DRC checks were carried out according to the 28 nm library requirements. The timing analysis implemented by the STA tool also included exhaustive signal integrity checks [28]. The difference in timing closure between all four physical design flows was less than 11 ps, which is less than 0.6% of the clock period. All flows were executed on a 64 bit Linux server (64 bit, 2.8 GHz with 64 GB RAM).

This section shows the comparison results for the four examined flows in terms of area, wire-length, static leakage, and runtime. First, we demonstrate the benefit of using the proposed improved SSRPG flow in terms of runtime. Then, we compare the proposed flow with the common SRPG and the no-retention flows. Table 2 depicts the comparison between the improved localized SSRPG flow, which uses the unique placement constraint rules, the common SRPG, and the distributed SSRPG physical design flows. It is shown that applying the extra placement rules, with regards to the selected retention FF's, improves the place and route Encounter tools' runtime by 11% compared to the distributed SSRPG and by 23% compared to the conventional SRPG flow. This is a considerable improvement compared to the runtime of the distributed flow, which does not apply any specific placement rules regarding the retention cells. The major improvement is achieved in the placement stage, in which the runtime is decreased by 29% compared to the distributed SSRPG flow. This is a significant result since the placement stage is an iterative stage due to the floorplan area estimation process. Moreover, the improved localized proposed flow outperforms the conventional SRPG by 63% in terms of placement runtime. The runtime for the routing stage is improved by 8% and 9% compared to the distributed SSRPG and SRPG, respectively. The runtime for the CTS stage is improved by 13% compared to the SRPG flow. Table 3 depicts the comparison between the four examined flows in terms of area, design density, number of library cells, wire-length, static leakage, and back-end tools runtime. As expected, the required area for SRPG implementation is 20% larger compared to the no-retention case. The implementation of the SSRPG approach results in a 16% area saving factor compared to SRPG. Moreover, almost no extra area is required for implementing the SSRPG flow compared to the no-retention case. While the wire length for SRPG is significantly larger compared to the no-retention flow, with about a 12% wiring increase, both SSRPG flows require only about 4% extra wiring compared to the no-retention case. This additional wiring overhead is required for connecting the retention cells to the non-gated power supply and power-gating controls. The increased wire-length induced by gathering all retention flip-FFs in a localized region is less than 1% compared to the distributed SSRPG.

Table 2. Place and Runtime Routing Comparison.

Run-Time (Hours)	No Retention	SRPG	Distributed SSRPG	Localized SSRPG
Placement	9.11	11.42	6.02	4.27
CTS	9.85	6.15	5.53	5.32
Routing	14.75	27.13	26.83	24.63
Total	33.71	44.7	38.38	34.22

Table 3. Physical design flow Comparison.

Physical Design Parameter	No Retention	SRPG	Distributed SSRPG	Localized SSRPG
Macro area (mm^2)	0.594	0.716	0.600	0.600
Design density (%)	72.2%	69.7%	72.3%	70.0%
Total library cells	315,837	318,052	313,679	309,369
Wire-length (m)	6.561	7.319	6.833	6.887
Static leakage (mW)	34.62	2.213	0.085	0.085
Backend Run-time (Hours)	33.72	44.7	38.38	34.22
Retained FFs	0	61,944	2522	2522

The increasing wiring can explain this since the retention FFs are associated along with other non-retention FFs. However, this wire-length is compensated due to the reduced distance between the retention cells to the always-on PDN and to the retention controls in the improved SSRPG flow. Table 3 shows that although the macro area is the same for both SSRPG flows, the design density (as measured by the Encounter Cadence tool) is reduced by 2.3% for the improved localized SSRPG compared to the distributed SSRPG. The lower density hints towards a lower crosstalk, though this still needs to be proved using bespoke benchmarks. Therefore, a better immune to crosstalk effects might be achieved using the localized PD approach. Spice simulations show that for both PD flows, the used gridlines meet the IR drop worst-case conditions (according to TSMC 28 nm library).

This can be explained due to the better routeability achieved by limiting the retention power grid to a specific localized region and therefore reducing the area occupied by both the always-on PDN and the retention control wiring. A significant improvement is also demonstrated for the static power leakage. Although SRPG reduces the static power leakage by 94% compared to the no-retention flow (whereas the supplies are always on), both SSRPG flows reduce the static power leakage by 99.7%. It is also important to notice that SSRPG outperforms the SRPG flow by 96% in terms of static leakage.

The efficiency of the improved SSRPG approach is expressed by the significant improvement in terms of back-end runtime. The required runtime for implementing the place and route stages is compared. While SRPG increases the runtime by a significant factor of 33%, the improved SSRPG flow can be implemented with a negligible overhead of only 3% compared to the non-retention flow. Moreover, the speed up comparing to the distributed SSRPG flow is about 11%. It should be noted that the improved SSRPG outperforms the distributed SSRPG in terms of back-end runtime in spite of the slightly increased wire length. This can be explained by the lower design density in the case of improved SSRPG due to the reduced buffers (as indicated by the total library cells) required to support the specific clock-tree for the retention controls compared to the distributed SSRPG flow.

4. Summary and Conclusions

This work presents a novel approach for SoC physical design implementation based on Selective State Retention techniques. The additional wiring required for the extra power grid network for the retention cells and power-gating controls for the state retention logic increases the complexity of the physical design and directly affects the tools' runtime and the ability to converge for large designs. Therefore, this work investigates the effect of the selective approach on the complexity of the physical design implementation and proposes a unique flow to efficiently address SoC design based on selective state retention techniques. We demonstrate a significant reduction of the metal area required for the extra power supply network using the proposed approach. This is done by applying some unique placement rules to the physical design implementation flow utilizing the selectivity feature. This results in optimal cell placement and power grid allocation, which significantly increase the potential routing area, directly improving the convergence time of the Place and Route tools. Furthermore, it is shown that reducing the extra power supply network area also leads to a significant reduction of the runtime required for the placement tools.

We also compare the SRPG and SSRPG physical design implementations in terms of power, area, wire-length, and physical design tools runtime and quantify the area and runtime saving factors result from selectivity. Experimental results show that implementing the SSRPG approach using the proposed physical design flow yields an area-saving factor of 16% compared to SRPG, which is in accordance with the previously estimated factor reported in recent publications. Furthermore, the static leakage is decreased by 96% compared to SRPG and is negligible compared to no retention. Tool complexity overhead was also reduced as such that the runtime overhead was negligible compared to the no retention physical design flow. Finally, by applying certain placement rules for the retention cells, the tool runtime for the improved SSRPG was further reduced by 11% compared to the common SSRPG and by 23% compared to SRPG.

The proposed improved localized SSRPG flow facilitates the complexity of the physical design implementation for retention-based design. This approach leads to both reducing the number of metal layers used for the always-on power distribution and therefore facilitates the signals routing, and reducing the wiring used for retention control signals as well as simplifying the isolation of the always-on domain from the power-gated domain. As a result, the runtime of the place and route tools is significantly reduced due to the wiring complexity reduction.

Moreover, to the best of our knowledge, this is the first work that demonstrates and quantifies the benefit of applying the SSRPG approach in real physical design implementation and demonstrating actual area, power, and tools runtime saving factor.

Author Contributions: Both authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Austin, T.; Baauw, D.; Mudge, T.; Flautner, K.; Hu, J.S.; Irwin, M.J.; Kandemir, M.; Narayanan, V. Leakage current: Moore's law meets static power. *IEEE J. Comput.* **2003**, *36*, 68–75.
2. Horiguchi, M.; Sakata, T.; Itoh, K. Switched-source-impedance CMOS circuit for low standby sub-threshold current giga-scale LSI's, Solid-State Circuits. *IEEE J.* **1993**, *28*, 1131–1135.
3. Zhigang, H.; Buyuktosunoglu, A.; Srinivasan, V.; Zyuban, V.; Jacobson, H.; Bose, P. Micro architectural techniques for power gating of execution units. In Proceedings of the International Symposium on Low Power Electronics and Design, ISPLED, Newport Beach, CA, USA, 9–11 August 2004; pp. 32–37.
4. Henry, M.B. *Emerging Power-Gating Techniques for Low Power Digital Circuits*; Virginia Polytechnic Institute and State University: Blacksburg, VA, USA, 2011.
5. Weihan, W.; Ohta, Y.; Ishii, Y.; Usami, K.; Amano, H. Tradeoff analysis of fine-grained power gating methods for functional units in a CPU. In Proceedings of the Cool Chips XV (COOL Chips), Yokohama, Japan, 18–20 April 2012; pp. 1–3.
6. Henry, M.B.; Nazhandali, L. Design techniques for functional-unit power gating in the Ultra-Low-Voltage region. In Proceedings of the Design Automation Conference (ASP-DAC), Sydney, NSW, Australia, 30 January–2 February 2013; Association for Computing Machinery: New York, NY, USA, 2012; pp. 609–614.
7. Dasnurkar, S.; Datta, A.; Abu-Rahma, M.; Nguyen, H.; Villafana, M.; Rasouli, H.; Tamjidi, S.; Cai, M.; Sengupta, S.; Chidambaram, P.R.; et al. Experiments and analysis to characterize logic state retention limitations in 28 nm process node. In Proceedings of the IEEE 31st VLSI Test Symposium, VTS, Berkeley, CA, USA, 29 April–2 May 2013; pp. 1–6.
8. Henzler, S.; Nirschi, T.; Pacha, C.; Spindler, P.; Teichmann, P.; Fulde, M.; Fischer, J.; Eireiner, M.; Fischer, T.; Georgakos, J.; et al. Dynamic state-retention flip flop for fine-grained sleep-transistor scheme. In Proceedings of the European Solid-State Circuits Conference, ESSCIRC, Grenoble, France, 12–16 September 2005; pp. 145–148.
9. Shigematsu, S.; Mutoh, S.; Matsuya, Y.; Tanabe, Y.; Yamada, J. A 1-V high-speed MTCMOS circuit scheme for power-down application circuits. *IEEE J. Solid-State Circ.* **1997**, *32*, 861–869. [CrossRef]
10. Le-Coz, J.; Flatresse, P.; Clerc, S.; Belleville, M.; Valentian, A. 65 nm PD-SOI glitch-free Retention Flip-Flop for MTCMOS power switch applications. In Proceedings of the IEEE International Conference on IC Design & Technology, ICICDT, Kaohsiung, Taiwan, 2–4 May 2011; pp. 1–4.
11. Chul-Moon, J.; Kwan-Hee, J.; Eun-Sub, L.; Minh, V.H.; Kyeong-Sik, M. Zero-Sleep-Leakage Flip-Flop Circuit with Conditional-Storing Memristor Retention Latch. *IEEE Trans. Nanotechnol.* **2011**, *11*, 360–366.
12. Kyungho, R.; Jisu, K.; Jiwan, J.; Kim, J.P.; Kang, S.H.; Seong-Ook, J. A Magnetic Tunnel Junction Based Zero Standby Leakage Current Retention Flip-Flop. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2011**, *20*, 2044–2053.
13. Jung-Hyun, P.; Heechai, K.; Dong-Hoon, J.; Kyungho, R.; Seong-Ook, J. Level-Converting Retention Flip-Flop for Reducing Standby Power in ZigBee SoCs. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2014**, *23*, 413–421.
14. Ting-Wei Chiang, C.; Kai-Hui Chang, C.; Yen-Ting, L.; Jiang, J.-H.R. Scalable sequence-constrained retention register minimization in power gating design. In Proceedings of the Design Automation Conference, DAC, San Francisco, CA, USA, 7–11 June 2015; Association for Computing Machinery: New York, NY, USA; pp. 1–6.
15. Darbari, A.; Hashimi, B.M.A.; Flynn, D.; Biggs, J. Selective state retention design using symbolic simulation. In Proceedings of the 2009 Design, Automation and Test in Europe Conference and Exhibition, Nice, France, 20 April 2009; IEEE: Piscataway, NJ, USA; pp. 1644–1649.
16. Greenberg, S.; Rabinowicz, J.; Tsechanski, R.; Paperno, E. Selective State Retention Power Gating Based on Gate-Level Analysis. *IEEE Trans. Circ. Syst. I* **2013**, *61*, 1095–1104. [CrossRef]
17. Greenberg, S.; Rabinowicz, J.; Manor, E. Selective State Retention Power Gating Based on Formal Verification. *IEEE Trans. Circ. Syst. I* **2014**, *62*, 807–815. [CrossRef]
18. Wen-Hsiang, C.; Chao, M.C.-T.; Shi-Hao, C. Practical Routability-Driven Design Flow for Multilayer Power Networks Using Aluminum-Pad Layer. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2013**, *22*, 1069–1081.
19. Hyung-Ock, K.; Youngsoo, S. Semicustom Design Methodology of Power Gated Circuits for Low Leakage Applications. *IEEE Trans. Circ. Syst. II Express Briefs* **2007**, *54*, 512–516.
20. Seomun, J.; Youngsoo, S. Design and Optimization of Power-Gated Circuits with Autonomous Data Retention. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2009**, *19*, 227–236. [CrossRef]
21. Golshan, K. Physical Design Essentials. In *An ASIC Design Implementation Perspective*, 1st ed.; Springer: New York, NY, USA, 2007.
22. Flynn, D.; Aitken, R.; Gibbons, A.; Kaijian, S. Low Power Methodology Manual. In *For System-on-Chip Design*, 1st ed.; Springer: New York, NY, USA, 2007.
23. Mahmoodi-Meimand, H.; Roy, K. Data-retention flip-flops for power-down applications. In Proceedings of the 2004 IEEE International Symposium on Circuits and Systems, Vancouver, BC, Canada, 23–26 May 2004; pp. 1–4.
24. Henry, M.B.; Nazhandali, L. NEMS-Based Functional Unit Power-Gating: Design, Analysis, and Optimization. *IEEE Trans. Circ. Syst. I* **2013**, *60*, 290–302. [CrossRef]

25. Wang, K.; Marek-Sadowska, M. On-chip power-supply network optimization using multigrid-based technique. *IEEE Trans. Comput.-Aided Des. Integr. Circ. Syst.* **2005**, *24*, 407–417. [CrossRef]
26. Ye, T.T.; de Micheli, G. Data path placement with regularity. In *Proceedings of the ACM/IEEE Computer Aided Design, ICCAD*; San Jose, CA, USA, 5–9 November 2000, IEEE: Piscataway, NJ, USA; pp. 264–270.
27. Zhaohui, H.; Pierres, A.; Hu, S.; Chen, F.; Royannez, P.; Pek, S.E.; Ling, H.Y. Practical and efficient SOC verification flow by reusing IP testcase and testbench. In Proceedings of the SoC Design Conference, ISOCC, Jeju, Korea, 4–7 November 2012; pp. 175–178. Available online: http://www.isocc.org (accessed on 26 July 2021).
28. Bhasker, J.; Chadha, R. Static Timing Analysis for Nanometer Designs. In *A Practical Approach*, 1st ed.; Springer: New York, NY, USA, 2009.
29. Gunnala, V. Choosing Appropriate Utilization Factor and Metal Layer Numbers for an Efficient Floor Plan in VLSI Physical Design. *Int. J. Eng. Res. Appl. IJERA* **2012**, *2*, 456–462.
30. Voldman, S.H. *Latchup*, 1st ed.; Wiley: West Sussex, UK, 2007.