2025

Documentation of Power Planning

DATE: 6TH FABRUARY, 2025

PREPARED BY AMIT SAHA

Table of Contents

Power Planning

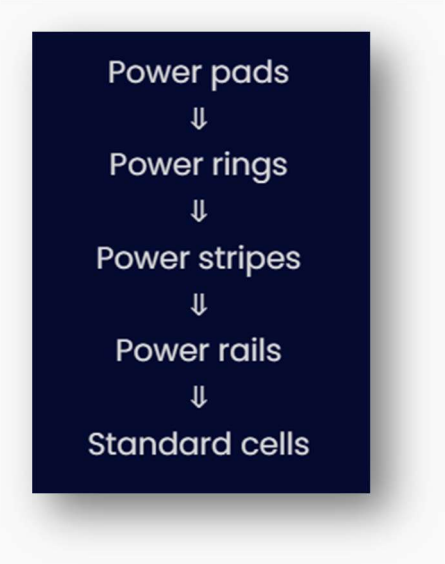
Introduction

Before diving into power planning in ASIC designs, let's consider why it's necessary. In higher node technologies, power planning wasn't a major concern; the focus was more on area and performance. However, as technology advanced to lower nodes, power considerations began to overshadow area and performance. This shift is largely due to the increasing prevalence of portable, battery-powered devices like TWS earbuds, smartwatches, and fitness bands. These devices require small batteries to last longer, making efficient power management crucial.

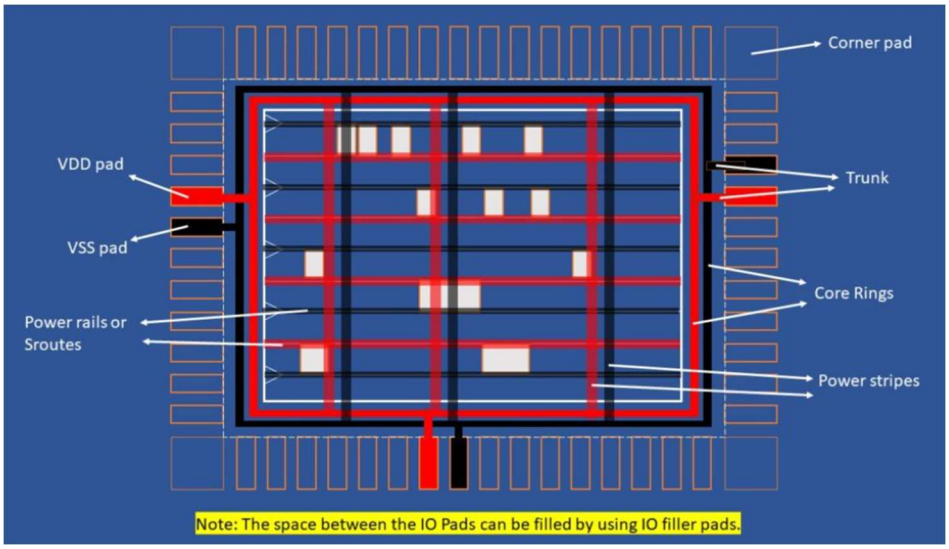
The miniaturization of chips brings significant power management challenges. One major issue is heat buildup. As more transistors are packed into a smaller area, power density increases, generating more heat in a confined space and creating hot spots that can damage the chip. Additionally, transistors require adequate thermal headroom to function properly.

Another challenge is maintaining signal integrity. Uneven power delivery across the chip can cause voltage fluctuations, which act like electrical bumps for signals traveling across the chip, making them unreliable and prone to errors. Chip designers must ensure a smooth and consistent power flow to maintain reliable signal transmission.

Power planning is a crucial step in the ASIC design process, particularly during the floor-planning phase. The primary goal of power planning is to establish an effective method for delivering power from external sources to the standard cells and macros through the power routes. During this stage, a power grid network is created to ensure uniform power distribution to all components of the chip. This involves providing sufficient power to macros, standard cells, and other cells within the design. Additionally, effective power planning helps in minimizing power noise and ensuring the reliability and performance of the chip. It also involves considering power integrity and thermal management to prevent overheating and ensure the chip operates efficiently under various conditions.
The flow of power or we can say levels of power distribution shown in below flow:



This entire process is known as the Power Distribution Network (PDN). The PDN is responsible for delivering power from external sources to the standard cells within the chip. The following figure illustrates an example of a PDN.



Types of Power Management

There are two types of power planning management. They are:

Core Cell Power Management

This type focuses on the internal components of the chip, such as macros and standard cells. Key elements include:

- **Power Rings**: VDD (power) and VSS (ground) power rings are formed around the core and macros to ensure stable power delivery.
- **Power Straps and Trunks**: Horizontal and vertical power straps and trunks are created to distribute power uniformly across the core area.
- **Power Mesh**: A network of power lines that helps in distributing power effectively to all parts of the core, minimizing IR drops and ensuring reliable operation.

I/O Cell Power Management

This type deals with the power distribution to the input/output cells of the chip. Key elements include:

- **Power Rings for I/O Cells**: Similar to core cells, power rings are formed around I/O cells to provide stable power.
- **Trunks Between Core and I/O Cells**: Power trunks are constructed to connect the core power ring with the power pads, ensuring efficient power transfer.
- **Power Pads**: These are the points where the external power supply connects to the chip, and they need to be designed to handle the required current and voltage levels.

The goals for Power Plan

Effective power planning in VLSI addresses these challenges by establishing a robust power delivery network that:

- **Delivers Power Evenly**: Ensures that every transistor receives the necessary voltage for consistent performance across the chip.
- **Minimizes Voltage Drops**: Power planning minimizes resistance in the power delivery network, ensuring sufficient voltage reaches all parts of the chip.
- **Avoids Electromigration**: Higher metal layers with lower resistance are chosen to supply power to the block, reducing the risk of electromigration. The width of the metal layer is decided based on the electromigration limit.
- **Prevents Overheating**: By controlling current flow, power planning reduces the risk of metal wires weakening due to excessive current, thereby preventing overheating.

Inputs for Power Plan

- **Floorplan Database:** This database contains the layout and architecture of the circuit, enabling designers to define how components will be placed on the chip.
- **Power Rings and Power Straps**: Designers must specify the width and configuration of power rings and straps to deliver VDD and VSS (power and ground) across the chip.
- **Spacing Specifications:** The spacing between VDD and VSS straps must be determined to ensure efficient power distribution without creating too much electrical noise.
- **Current Ratings and Power Consumption Estimates:**
- Inputs related to the total current drawn by the chip and estimates of power consumption (both static and dynamic) are critical to designing a suitable power grid network.
- **Power Pins Requirements:** The number of power pins required for the design must be calculated based on the power needs of the chip as well as the specifics of the layout.
- **Power Mesh Configuration:** The design must include a power mesh consisting of horizontal and vertical lines on the chip that help distribute power effectively.
- **IR Drop Budget:** A budget for IR (voltage drops due to resistance) drop must be established to ensure that the design maintains adequate voltage levels under operational conditions. This involves methods for calculating and improving both static and dynamic IR drops.
- **Power Management Strategies:** Definitions of strategies for managing power distribution within core cells and I/O (Input/Output) cells, including the design of separate power rings for power-critical macros.

Output of Power Planning

- Design incorporating a power structure
- Floorplan featuring a synthesized power mesh

Power Planning Calculation

- Calculating the total number of core power pads required.
- Calculating the core current based on the core power and voltage.
- Calculating the total power (dynamic and static).
- Determining the total number of power pads and power pins.
- Calculating the core PG ring width.

Sanity Checks after Power planning

- Ensure all cells receive power.
- Fix any power/ground (PG) opens or shorts in the design.
- Perform power design rule checks (DRCs).
- Assess the impact of IR drop on the design's PG grid and enhance the grid to achieve an acceptable IR drop.

Terminologies

Power Pad

A power pad is a crucial component in ASIC designs. It is placed alongside I/O pads in the pad section of the chip. The primary function of a power pad is to act as a bridge, transferring power from external sources to the power rings within the chip. This ensures that the internal components receive the necessary power to operate efficiently.

Trunks

In ASIC designs, trunks are the nets that connect the power pads to the core rings. They play a crucial role in ensuring that power is efficiently distributed from the external power sources to the core of the chip. This connection is vital for maintaining the integrity and reliability of the power distribution network (PDN) within the chip.

Core Rings

In ASIC designs, core rings are essential components of the power distribution network (PDN). These rings, which surround the core logic of the chip, consist of power (VDD) and ground (VSS) rings. They receive power from the power pads and distribute it to the power stripes, ensuring a stable and uniform power supply across the chip. This helps in reducing voltage drops and maintaining signal integrity throughout the design.
Note: VDD ring + VSS ring = one core ring set.

Stripes:

Power strips (or stripes) are narrow metal lines that run across the chip, connecting the core rings to the standard cells and macros. These strips ensure that power is evenly distributed throughout the chip, helping to maintain a stable voltage and reduce the risk of voltage drops and power noise. They play a critical role in the overall power distribution network (PDN), ensuring reliable operation of the chip's components. Power stripes are always created in the top metal layer.

Note: VDD + VSS = one stripe set

Power Rails (Sroutes)

In ASIC designs, power rails are metal lines that run horizontally across the standard cells, typically in the first metal layer (Metal-1). They are connected to the power stripes via power vias and are responsible for delivering power directly to the standard cells. Power rails ensure that each cell receives a stable and consistent power supply, which is crucial for the reliable operation of the chip.

Static Power Dissipation

Static power dissipation occurs when a circuit consumes power even when it is not switching, meaning there is no change in input and the clock is not applied. This type of power dissipation is primarily due to leakage currents that flow through transistors even when they are in an off state. As technology nodes shrink, managing static power dissipation becomes increasingly important to ensure the efficiency and longevity of battery-powered devices. The following list is the leakage currents in the MOSFETs:

- Sub-threshold conduction – When $V_{gs} <= V_{th}$, leakage current flows from drain to source.
- Reverse bias pn junction – In NMOS, the drain is N-type and the bulk is P-type. So, drain is at a higher potential than bulk. Due to this some leakage current flows from the drain to the bulk.
- Gate tunneling – Due to the gate thickness, some leakage current flows from the gate to the bulk.

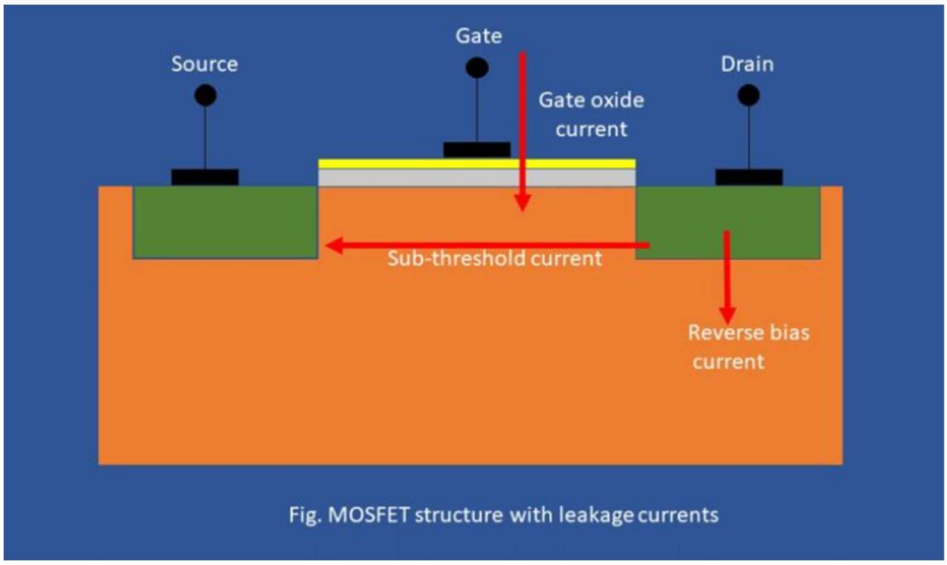


Fig. MOSFET structure with leakage currents

Dynamic Power Dissipation

Dynamic power dissipation occurs when a circuit is actively switching states. It is primarily caused by the charging and discharging of load capacitances during the transition of logic states. Dynamic power dissipation happens for two reasons. They are:

- Switching power
- Short-circuit power

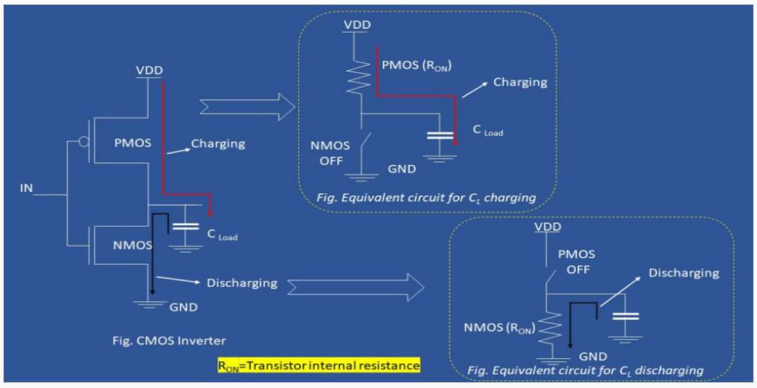
Switching Power

When the circuit switches from logic 0 to logic 1, power is drawn from VDD, passing through the PMOS and charging the load capacitor along with other parasitic capacitors. The switching power depends on the frequency at which the load capacitor is charged and discharged.

$$P_{switching}=\alpha \cdot C_{load} \cdot Vdd^2 \cdot f$$

Where:

α is the activity factor, representing the fraction of gates switching,
C_{load} is the load capacitance,
Vdd is the supply voltage,
f is the clock frequency.



Switching power is a significant concern in modern ASIC designs, especially as operating frequencies and transistor densities increase. Techniques to reduce switching power include lowering the supply voltage, reducing capacitance, and minimizing the switching frequency.

Short Circuit Power

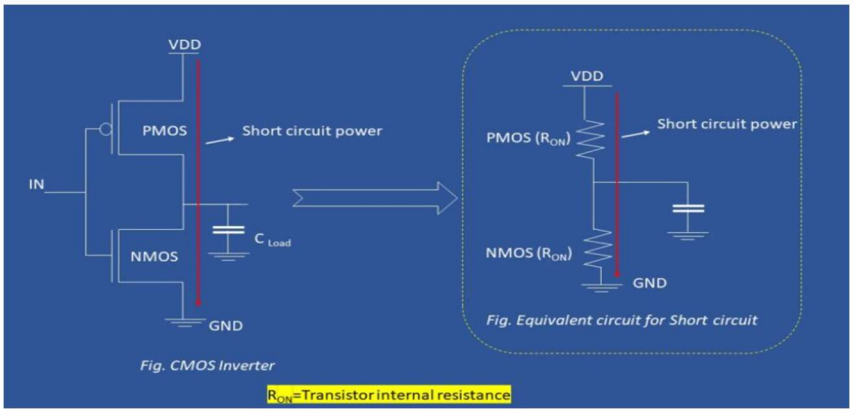
Short circuit power is a type of power dissipation that occurs during the switching of CMOS circuits. When the input of a CMOS circuit transitions from logic 0 to logic 1 (or vice versa), there is a brief period during which both the PMOS and NMOS transistors are simultaneously on. This creates a direct path between the power supply (VDD) and ground (VSS), resulting in a short circuit current. The duration and slope of the input signal, as well as the sizes and characteristics of the transistors, influence the magnitude of short circuit power. Managing short circuit power is crucial for reducing overall power consumption and ensuring the efficiency of the circuit.

We strive to be the most trusted partner for the world's hardest engineering problems.

$$P_{short\text{-}circuit} = I_{sc} \cdot V_{dd} . t_{sc}$$

Where:

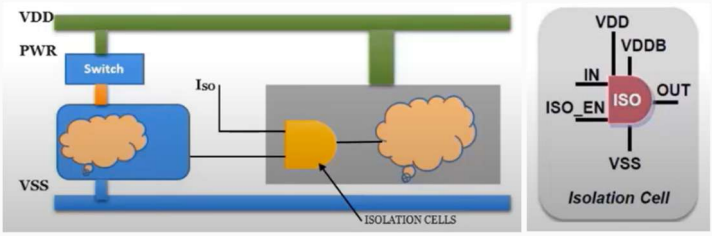
Isc is the short circuit current,
V_{dd} is the supply voltage,
t_{sc} is transition time



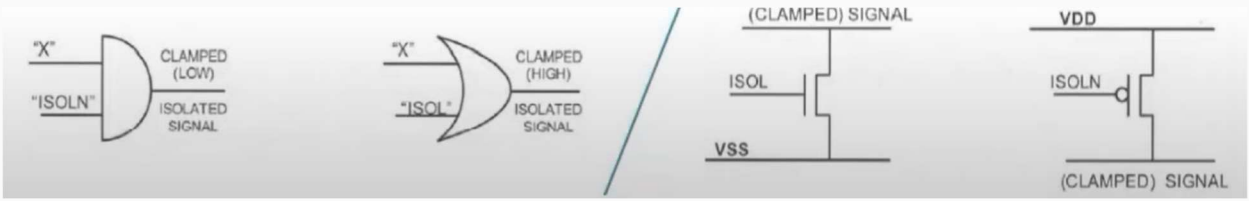
Isolation Cell

When power domains are turned off, their outputs no longer drive signals, resulting in floating nodes. These floating nodes can cause issues when active domains receive them as inputs, potentially leading to crowbar currents that disrupt the proper functioning of the powered-up domain.

Isolation cells, also known as "clamps," address this issue by maintaining the outputs of powered-off sub-blocks at predefined values. This ensures that the inactive sub-blocks do not interfere with the functionality of active sub-blocks.

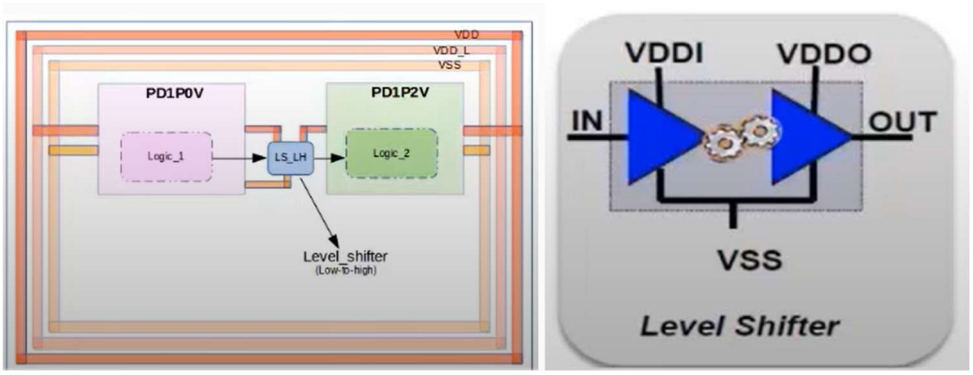


Isolation cells are continuously powered and can output 0, 1, or retain the previous value of a deactivated domain. During normal operation, these cells transmit logic values, but when a control signal is activated, they clamp the output to a specified value. Isolation cells can clamp the output of a powered-down block to a defined value ('0', '1', or the last value). There are two types of clamp cells: gate-type (AND, OR) and transistor-type (pull-up, pull-down).

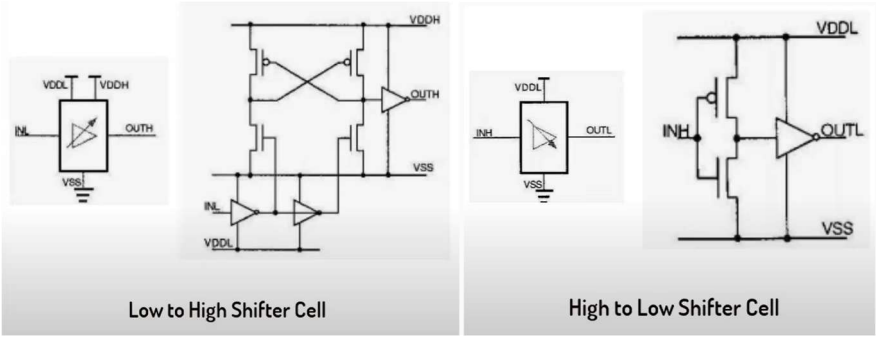


Level Shifter

Level shifters function similarly to buffers and are essential in low-power designs with multiple voltage domains or dynamic voltage scaling. They convert a logic value from one voltage domain to the same logic value in another voltage domain. An "Up" level shifter converts a logic value from a lower voltage domain to a higher one, while a "Down" level shifter converts a logic value from a higher voltage domain to a lower one.

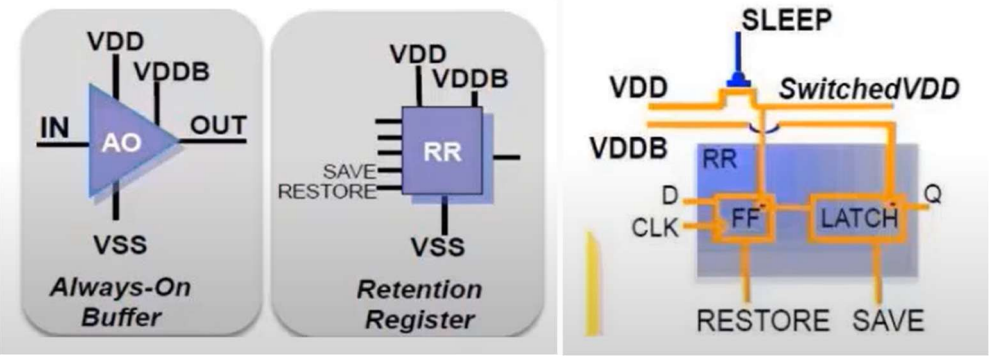


Level shifter cell will place near the receiver side.



Retention Cell

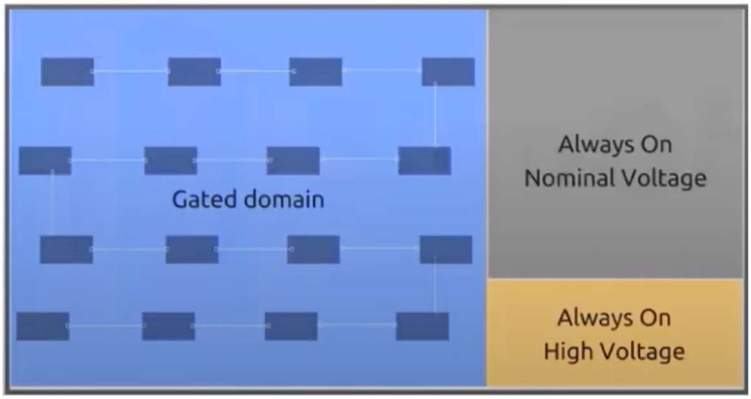
To reduce power consumption, memories are shut down by switching off their power domain or when they are not in use. When power is switched off, registers become corrupted, typically represented as 'X' (unknown). Some memories need to retain their values for a quick wake-up. In these cases, only the memory array remains powered on during the shutdown, while the peripheral interfaces are powered off.



Retention registers keep their previous active value after being shut down. Retention registers save state information before a power domain is switched off and restore it when the power is turned back on. Retention registers comprise of two circuits. Standard register logic, supplied by primary power VDD. Shadow latch retention circuitry, with alternate supply VDDB. SAVE – transfers FF content into shadow latch during shutdown. Restore – transfers state from shadow latch to FF when powered back.

Power Switches

Power switches are necessary to control the power supply of a gated domain when it is not needed. These switches are MT-CMOS (multi-threshold) cells, which have a very high threshold voltage when the device is off and a very low threshold voltage when the device is on. Power switches are integrated into the power mesh, supplying power to all gated domain cells through these switches. Therefore, a single or a few switches are insufficient. Instead, a robust network of power switches, connected in a daisy chain fashion, is incorporated into the design. Power switches are insert only with the physical only cells.



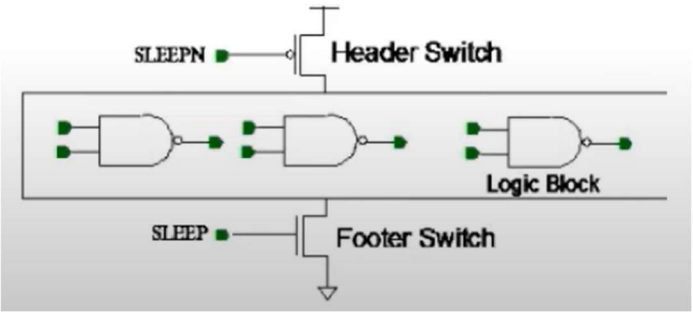
There are 2 types of power switches:

Header

Header switches, implemented using PMOS transistors, control the Vdd supply. PMOS transistors are less leaky than NMOS transistors of the same size. Header switches turn off Vdd while keeping Vss on, enabling a simple design with a pull-down transistor to isolate power-off cells and clamp output signals. This implementation typically consumes more area than a footer switch implementation.

Footer

Footer switches, implemented using NMOS transistors, control the Vss supply. The advantage of footer switches is their high drive capability, which results in a smaller area. However, NMOS transistors are leakier than PMOS transistors, making designs more sensitive to ground noise.



Always on Cell

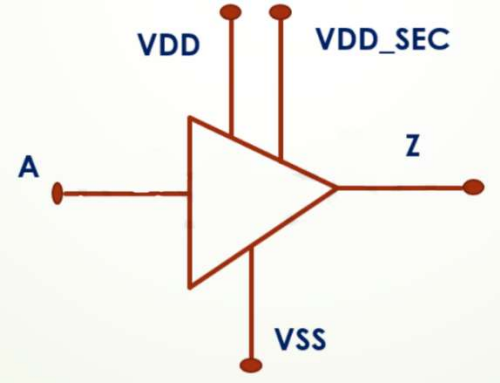
An always-on cell (AON) in VLSI is a digital circuit designed to remain powered and active even when the main power supply is turned off. These cells are commonly used in low-power applications like mobile devices, IoT devices, and other battery-powered electronics where certain functions must be maintained in low-power or standby mode.

AON cells handle tasks such as maintaining real-time clocks, monitoring external events, and preserving the device's state during sleep mode. They are also used in power management circuits to monitor battery status and shut down the device before the battery is completely drained.

These cells are typically built with specialized low-power logic gates and flip-flops that consume minimal power in standby mode. They also utilize power management techniques like voltage scaling, dynamic voltage and frequency scaling, and power gating to reduce power consumption.

Overall, AON cells are crucial for developing low-power, energy-efficient electronic devices and systems.

AON cells function like regular buffers but include an additional secondary always-on pin to keep the cells active even when the primary power is off in a domain. These cells remain 'on' in their region through a secondary backup supply pin, which provides the necessary current when the main power supply is unavailable.



IR Drop

The power supply in a chip is distributed through metal layers, which have finite resistance. When voltage is applied, current flows through these layers, causing a voltage drop known as IR drop. This drop can increase delay, violate timing, and degrade performance.
Acceptable IR drop values are set at the project's start and influence the derate value. If the IR drop exceeds the acceptable limit, the derate value must be adjusted to avoid optimistic timing calculations. For example, a design operating at 1.2V with a tolerance of ±0.4V must maintain voltage between 0.8V and 1.6V to ensure proper timing and functionality.

Power routes, which conduct significant current, are placed in higher metal layers to reduce resistance. Typically, the top two or three layers are reserved for top-level routing, while layers like Metal 11 and Metal 12 are used for power planning.
There are two types of IR drop:
1. Static IR drop
2. Dynamic IR drop

Static IR Drop

Static IR drop refers to the voltage drop in the power delivery network (PDN) when the circuit is in a static state, meaning no inputs are switching. This drop is primarily due to the resistance of the metal wires in the PDN and the average current flowing through them. The formula for static IR drop is:

$$V_{static\_drop} = I_{avg} \cdot R_{wire}$$

where:

- I_{avg} is the average current, which includes leakage currents,
- R_{wire} is the resistance of the metal wires.

Static IR drop can affect the performance of the chip by increasing delays and potentially causing functional failures due to timing violations. Managing static IR drop is crucial for ensuring the reliability and efficiency of the design.

Methods to improve static IR drop:

1. Go for higher layers if available
2. Increase the width of the stripes
3. Increase the number of wires
4. Check if any via is missing then add more via.

Dynamic IR Drop

Dynamic IR drop refers to the voltage drop in the power delivery network (PDN) caused by the high switching activity of transistors. This occurs when there is a sudden increase in current demand due to simultaneous switching of multiple circuits. Dynamic IR drop is influenced by the switching rate of the logic and can lead to temporary voltage drops that affect the performance and timing of the chip. Managing dynamic IR drop is crucial to ensure reliable operation and prevent timing violations.

Methods to improve dynamic IR drop:

1. Use de-cap cells.
2. Increase the number of stripes.

Electromigration

Electromigration (EM) is a phenomenon in which metal atoms in interconnects are displaced due to the momentum transfer from conducting electrons. This displacement occurs when high current densities cause metal ions to drift in the direction of electron flow. Over time, this can lead to the formation of voids and hillocks, which can cause open circuits or short circuits, ultimately affecting the reliability and performance of the chip.

Electromigration is particularly problematic in lower technology nodes where the cross-sectional area of metal interconnects is smaller. To mitigate EM, designers must carefully manage current density, optimize layout, and select appropriate materials. Techniques such as increasing the width of critical interconnects and using redundant vias can also help reduce the impact of electromigration.

Methods to solve EM:

1. Increase the width of wire
2. Buffer insertion
3. Downsize the driver
4. Switch the net to higher metal layers
5. Adding more vias
6. Keep the wire length short

Grid Structure

Power and ground nets are typically laid out on the metal layers. This involves creating power and ground structures for both IO pads and core logic. The power and ground buses for the IO pads are integrated into the pad itself and will be connected by abutment.

For core logic, a core ring encloses the core with one or more sets of power and ground rings. The next step is to construct the internal power and ground connections for the core logic, known as power and ground stripes. These stripes are repeated at regular intervals across the logic or specified region within the design. When these stripes run both vertically and horizontally at regular intervals, they form a power mesh.

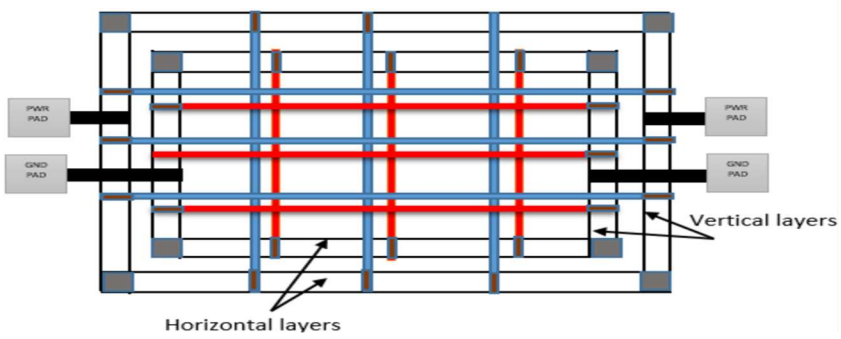


Fig: Ring and core power ground

The total number of stripes and their interval distance depend entirely on the ASIC core's power consumption. As power consumption (both static and dynamic) increases, the interval distance between power and ground straps also increases to reduce the overall voltage drop, thereby enhancing performance.

In addition to the core power and ground ring, macro power and ground rings should be created using vertical and horizontal metal layers. A macro ring encloses one or more macros, either completely or partially, with one or more sets of power and ground rings. It is highly recommended to verify power and ground connectivity after constructing the entire power and ground (PG) network.

Standard cells are placed in rows, side by side, touching each other. The horizontal routing across these cells is already completed, creating a continuous line. After optimization, if there are any gaps, filler cells are placed to fill these spaces.

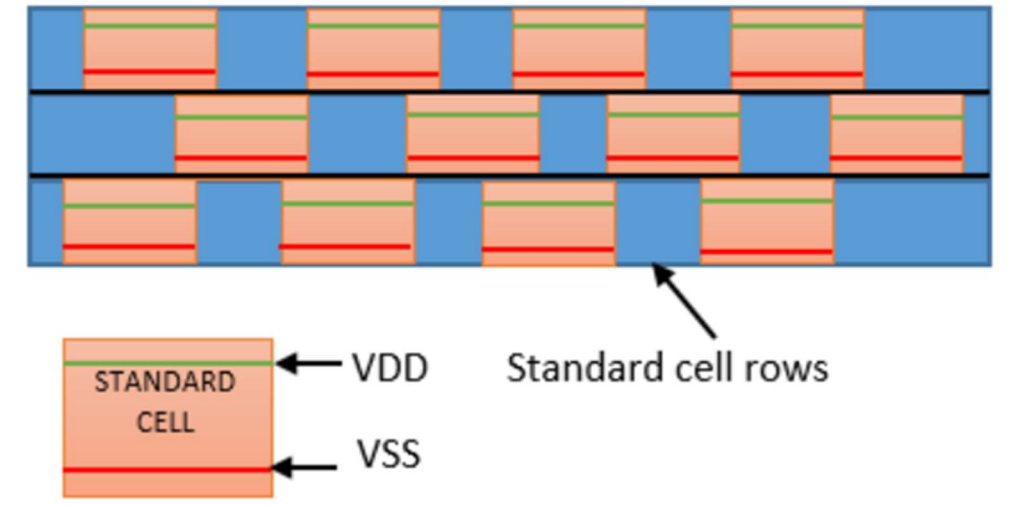


Fig: Standard cells are placed in the rows.

As we move to deep sub-micron technologies, the number of metal layers increases. For example, in 28 nm technology, there are only 7 metal layers, whereas in 7 nm technology, there are 13 metal layers. Additionally, the width of the layers increases from lower to higher metal layers. Therefore, it is better to use higher layers for power lines, as they can carry more current.

Low Power Design

Low power design encompasses a range of techniques and methodologies focused on minimizing both dynamic and static power consumption in integrated circuits (ICs). This approach is crucial for enhancing the energy efficiency and performance of electronic devices, especially in applications like mobile and battery-powered devices.

The goal of low power design is to minimize the individual components of power consumption, thereby reducing the overall power usage of an integrated circuit (IC). The power equation includes components for both dynamic and static power. Dynamic power consists of switching and short-circuit power, while static power is due to leakage current, which flows through the transistor even when there is no activity.

The value of each power component is influenced by several factors:

- Activity
- Frequency
- Transition time
- Capacitive load
- Voltage
- Leakage current
- Peak current

For instance, higher voltage results in higher power consumption by each component, leading to increased overall power. Conversely, lower voltage reduces overall power consumption. To achieve optimal performance with minimal power usage, trade-offs among these factors are explored and tested using various low power techniques and methodologies.

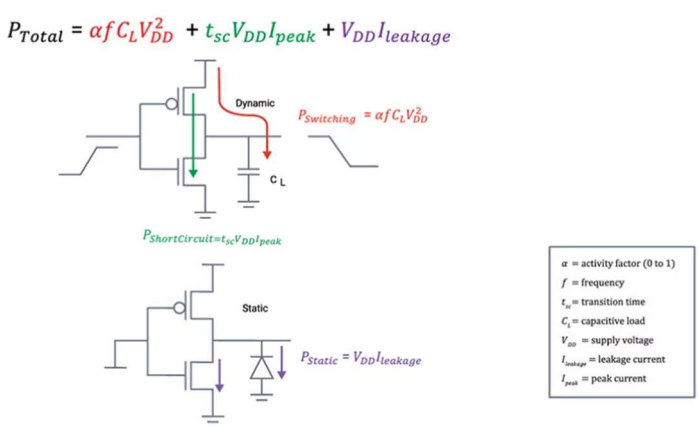


Figure 1: Power components and equation

Companies are constantly innovating to add new features and functionalities to portable, handheld, and battery-powered devices. For these products, extending battery life by reducing power consumption is a major advantage and highly valued by users. Additionally, minimizing the time it takes for a device to switch from OFF/SLEEP mode to ON/ACTIVE mode is crucial for providing a seamless user experience along with longer battery life.

For devices that are plugged in, power consumption remains important because it can influence the overall system cost by requiring additional cooling solutions, such as heat sinks, and increasing electricity expenses. In server farms, where large-scale parallel systems are used, reducing the power consumption of a single chip can lead to significant energy savings across the entire system. Upgrading these systems with newer, more energy-efficient integrated circuits (ICs) can result in substantial power and cost savings.

Low power Design Technique

There are numerous low power design techniques available, ranging from straightforward methods to more intricate and complex approaches. These techniques are essential for optimizing power consumption and enhancing the efficiency of integrated circuits (ICs).

Multi Voltage

This technique involves partitioning the functions of a chip based on performance characteristics. For example, one block may be high performance, requiring a higher voltage, while the rest of the chip operates at a lower performance level with a lower voltage. This approach saves power compared to designing the entire chip at a higher voltage, which is simpler but more power-intensive. By reducing voltage, both static and dynamic power components are decreased.

In multi-voltage designs, separate voltage islands are created, and voltage crossings between these islands may require "Level Shifter" (LS) cells. These cells help manage the different voltage characteristics of each block, ensuring proper functionality and performance.

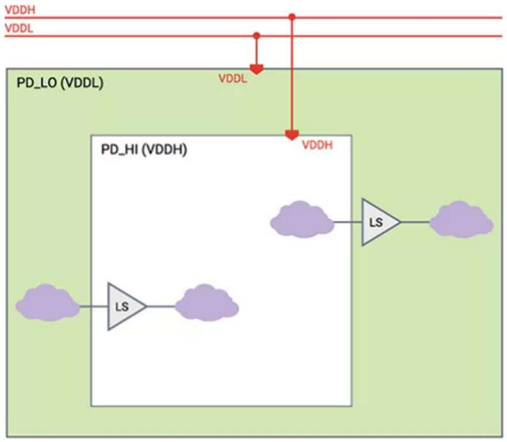
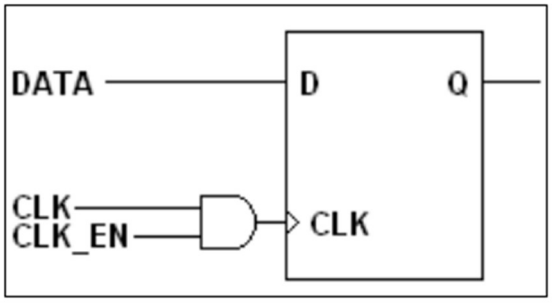
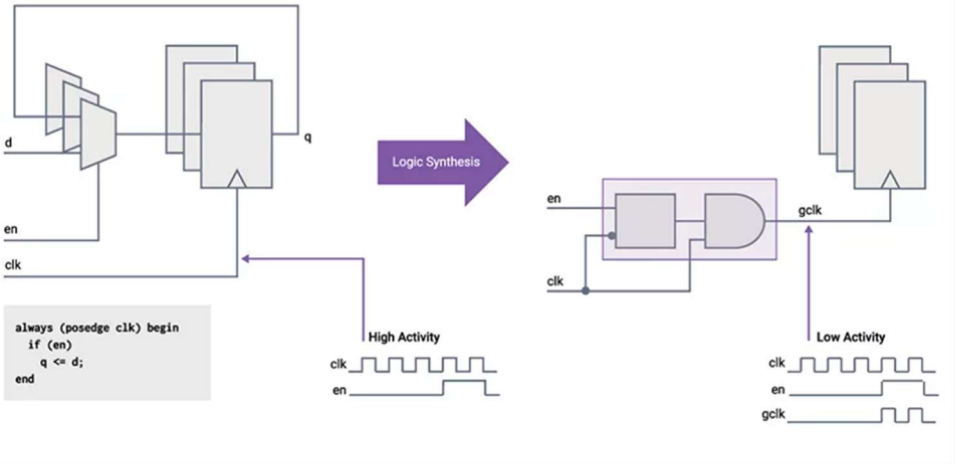


Fig: Multi Voltage

Clock Gating

Clock gating is a power-saving technique used in ASIC designs to reduce dynamic power dissipation. This technique is usually applied during logic synthesis, where enable flops are optimized into a clock gating structure. This optimization saves mux area and reduces the overall switching activity of the clock net. In terms of the power equation, the aim is to reduce capacitive load (through area reduction) and activity factors, thereby decreasing the switching power component of dynamic power. This method is straightforward and readily available for reducing power and area. However, it depends on the logic synthesis tool to perform this optimization. Fortunately, this technique is well-known and well-supported in most tools and design flows.





There are two main types of clock gating:

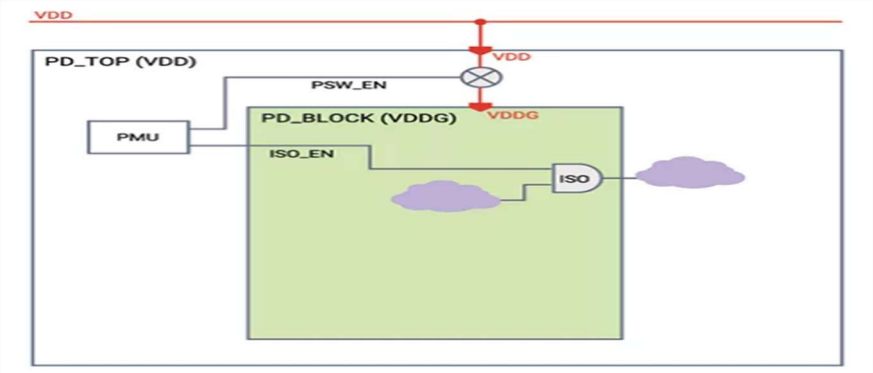
- **Intent-based clock gating**: Introduced during the RTL design phase, where specific conditions are defined to enable or disable the clock.
- **Tool-generated clock gating**: Implemented during synthesis, where tools identify and group flip-flops with similar control logic to enable clock gating.

By reducing unnecessary clock activity, clock gating helps in lowering overall power consumption without affecting the functionality of the design.

Power Gating

This technique involves partitioning functions on an IC, similar to multi-voltage designs, but with power supplies for the power domains connected to power switches. Power gating completely shuts off power to a block, reducing both static and dynamic power consumption by zeroing out the voltage when the block is off. This method offers significant power savings, making it ideal to shut off as many domains as possible, as frequently as possible, while maintaining functionality.

To achieve these power savings, power switches must be implemented in the design, along with isolation gates that clamp the boundaries of the power domain to known values when off. The power states of the design and the combination of ON/OFF states for given voltages must be carefully considered. Additionally, a power management unit (PMU) is required to control the power switch and isolation enable signals. It is crucial that the order of these signals is correct during power down and power up to ensure that values are clamped to the right levels at the right time during shutdown.



The Impact of Low-Power Design

The impact of low-power design goes beyond VLSI, promoting energy-efficient devices that help reduce carbon footprints, minimize electronic waste, and extend battery life. Efficient power usage also fuels technological innovation by removing energy constraints, enabling designers to develop and implement advanced applications that were previously limited by power demands. This approach is especially critical in the era of IoT, portable electronics, and large-scale data centers, where energy efficiency is essential for sustainable growth and operational efficiency.

Power Intent Information

Power intent in VLSI design is all about specifying how a chip manages power efficiently. Instead of hard-coding power details into the design, engineers use standardized formats like **UPF (Unified Power Format)** or **CPF (Common Power Format)** to define the power requirements and strategies. These formats help the design tools understand how to handle power efficiently across different parts of the chip.

Key Elements of Power Intent

Power Domains

Think of these as different zones in a chip, each with its own power settings. Some zones can stay active while others are powered down to save energy.

Power Rails

These are the voltage supply lines that feed different parts of the chip.

Voltage Islands

Similar to power domains, but focused on specific areas operating at different voltage levels to optimize energy use.

Power Modes

These define how different parts of the chip behave — whether they are fully active, in standby, or completely powered down.

Power Switches

Components that can turn power on or off for specific areas of the chip.

Retention Cells

Special circuits that remember critical information even when power is turned off.

Isolation Cells

These protect signals from getting messed up when some parts of the chip are powered off.

Level Shifters

Used to ensure that signals between different voltage regions communicate properly.

Clock Gating

A method to save power by turning off clock signals when they're not needed.

Why Power Intent Matters

- It helps reduce power consumption, which is crucial for battery-powered devices and data centers.
- Let's the chip adapt dynamically by powering down unused parts or adjusting voltage levels.
- Ensures low-power strategies work correctly during testing and simulation.
- Having a formal power intent makes it easier to use the design across different tools and environments.

CPF

CPF is designed to facilitate the specification of power-saving techniques early in the design process. It provides a standard format for describing various power domains and their supplies, enabling consistent implementation across different design steps.

It is designed by cadence in si2 standard. The use of CPF is power intent information can be given only once and can be used consistently by all tools. Aim of CPF is to support an automated power aware design infrastructure. It is strictly TCL based format.

CPF File Format Specifics

- It is strictly TCL based format.
- CPF file contains two categories of objects:
  - **Design objects:** These refer to the actual components within the design, such as modules, instances, nets, pins, and ports as they appear in the RTL (Register Transfer Level). Design objects are associated with the specification objects to implement the defined power management strategies.
  - **CPF objects:** These objects are created in CPF file. Such as analysis view, Isolation rule, level shifter rule, library set, power domain, power switch rule.

Analysis view: View that associates an operating corner with a power mode for which SDC constraints are specified.
Isolation rule: It defines the location and type of isolation logic to be added and condition for when to enable the logic.
Level Shifter rule: Defines the location and type of level shifter logic to be added.
Library set: Set of libraries having same set of operating conditions.
Power domain: Collection of instances that use same power supply during normal operation and can be switched on or off at the same time.

CPF Command

Command	Description	Example
set_cpf_version	It will set the version of CPF.	set_cpf_version 1.1
set_hierarchy_separator	Specifies the hierarchy delimiter character	set_hierarchy_separator /
define_library_set	It will define the timing library	define_library_set -name wc_0v81 -libraries {../LIBS/timing/library_wc_0v81.lib }
define_always_on_cell	Specifies the always on cells	define_always_on_cell -cells {PTLVLHLD* AOBUFF*} -power_switchable VDD -power TVDD -ground VSS
define_isolation_cell	Specifies the isolation cells	define_isolation_cell -cells { LVLLH* } -power VDD -ground VSS -enable NSLEEP -valid_location to
define_level_shifter_cell	Specifies level shifter cells	define_level_shifter_cell -cells { LVLHLD* } -input_voltage_range 0.72:0.81:0.09 -output_voltage_range 0.72:0.81:0.09 -direction down -output_power_pin VDD -ground VSS -valid_location to
define_power_switch_cell	Identify a power or ground switch cell	define_power_switch_cell -cells { HEADERHVT1 HEADERHVT2 } \ -stage_1_enable NSLEEPIN1 -stage_1_output NSLEEPOUT1 -stage_2_enable \ NSLEEPIN2 -stage_2_output NSLEEPOUT2 -type header -power_switchable VDD \ -power TVDD -stage_1_on_resistance 10 - stage_2_on_resistance 10

define_state_retention_cell	Specifies retention cell	define_state_retention_cell -cells { MSSRPG* } -cell_type \ master_slave -clock_pin CP -restore_check !CP -save_function !CP \ -always_on_components { DFF_inst } -power_switchable VDD -power TVDD \ -ground VSS
set_design	It will set the design	set_design top
create_operating_corner	It will create operating corner	create_operating_corner -name PMdvfs2_bc -voltage 0.88 -process 1 -temperature \ 0 -library_set bc_0v72
create_power_nets, create_ground_nets	It will create power and ground nets	create_power_nets -nets VDDL -voltage 0.72 -peak_ir_drop_limit 0 \ -average_ir_drop_limit 0 create_ground_nets -nets Avss -voltage 0.00 -peak_ir_drop_limit 0 \ -average_ir_drop_limit 0
create_power_domain	It will create power domain	create_power_domain -name PDshutoff_io -instances { IOPADS_INST/Pspifsip \ IOPADS_INST/Pspidip } -boundary_ports { spi_fs spi_data } \ -external_controlled_shutoff -shutoff_condition io_shutoff_ack

Please refer this following document to get basic CPF file:
Innovus User Guide -- CPF 1.1 Script Example

UPF

UPF is an IEEE standard developed to reflect a design's power intent at a relatively high level. UPF supports detailed descriptions of power-related functionalities, including when blocks are powered up to shut down and how voltage levels should be shifted between power domains. It is industry wide power format spec. to implement power techniques in design flow.

As power become an important factor there is a need for more systematic approach to reduce power in complex designs and UPF is developed to address this need. UPS scripts help describe power intent such as

- Power rails to be routed to individual blocks
- When blocks are expected to be powered up or shut down
- How voltage levels should be shifted between two different power domains.

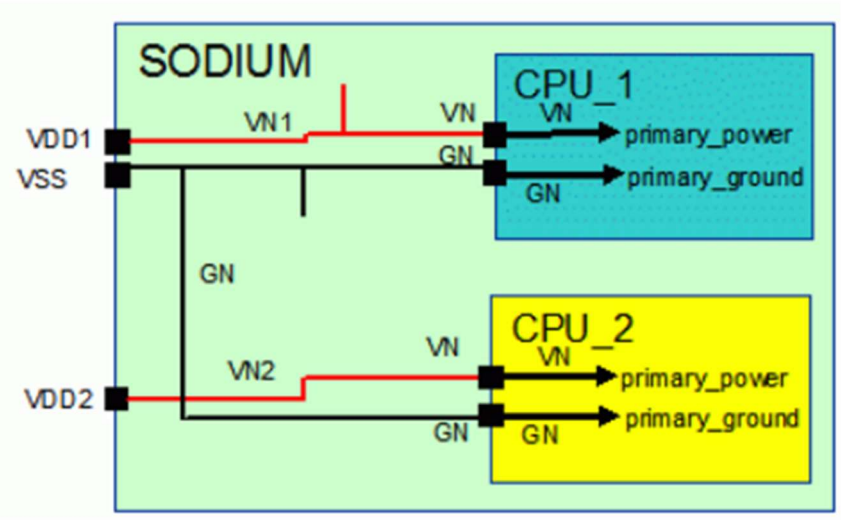
Power intent commands:

- Power state can be defined for a supply set, a power domain, a group, a model by using the command add_power_state.
  Syntax:
  add_power_state [-supply|-domain|-group|-model|-instance] object_name
  Object name is the simple name o an object. It can be the name of the supply set or power domain or group name or model name or instance name based on option specified.
- Supply net can be connected to the ports by using the command connect_supply_net.
  Syntax: connect_supply_net net_name –ports port_list

- Supply sets can be connected to the ports by using the command connect_supply_set.
  Syntax: Connect_supply_set supply_set_ref –elements element_list
- Logic net can be connected to logic ports by using the command connect_logic_net.
  Syntax: connect_logic_net net_name –ports port_list
- To execute the commands from the specified UPF file in the current scope or in the scope of each specified instance we need to use load_upf_command.
  Syntax: load_upf upf_file_name
  upf_file_name is the upf file that has upf commands.
- To change the current scope to the specified scope and to return the name of the previous scope we need to use set_scope command.
  Syntax: set_scope instance
  Instance that becomes current scope upon completion of the command.

UPF Example

Example-1: Multi-Voltage design



1. **Bottom-Up Power Intent Specification:**

   In a bottom-up flow, the lower-level CPU blocks are designed independently from the top-level chip. Therefore, their power intent must be specified at the block level and then later integrated into the chip-level power intent specification.

This is the power intent script at the block level, CPU.upf:

```
create_power_domain PD
create_supply_net VN -domain PD
create_supply_net GN -domain PD
set_domain_supply_net PD -primary_power_net VN -primary_ground_net GN
create_supply_port VN
create_supply_port GN
connect_supply_net VN -ports {VN}
connect_supply_net GN -ports {GN}
```

This is the power intent script at the top level, SODIUM.upf:

```
load_upf CPU.upf -scope CPU_1
load_upf CPU.upf -scope CPU_2
# still at scope SODIUM
create_supply_port VDD1
create_supply_port VDD2
create_supply_port VSS
create_power_domain PD
create_supply_net VN1 -domain PD
connect_supply_net VN1 -ports {VDD1 CPU_1/VN}
create_supply_net VN2 -domain PD
connect_supply_net VN2 -ports {VDD2 CPU_2/VN}
create_supply_net GN -domain PD
connect_supply_net GN -ports {VSS CPU_1/GN CPU_2/GN}
set_domain_supply_net PD -primary_power_net VN1 -primary_ground_net GN
# PD, CPU_1/PD and CPU_2/PD are different power domains.
```

2. **Top-Down Power Intent Specification:**

If the chip is designed from the top down, the power intent of the design can be specified directly from the top level, if desired, as demonstrated by the following example:

```
create_power_domain PD_CPU_1 -elements {CPU_1}
create_power_domain PD_CPU_2 -elements {CPU_2}
create_power_domain PD_SODIUM
create_supply_port VDD1
create_supply_port VDD2
create_supply_port VSS
create_supply_net VN1 -domain PD_CPU_1
create_supply_net VN1 -domain PD_SODIUM -reuse
connect_supply_net VN1 -ports {VDD1}
create_supply_net VN2 -domain PD_CPU_2
connect_supply_net VN2 -ports {VDD2}
create_supply_net GN -domain PD_CPU_1
create_supply_net GN -domain PD_CPU_2 -reuse
create_supply_net GN -domain PD_SODIUM -reuse
connect_supply_net GN -ports {VSS}
set_domain_supply_net PD_CPU_1 \
 -primary_power_net VN1 -primary_ground_net GN
set_domain_supply_net PD_CPU_2 \
 -primary_power_net VN2 -primary_ground_net GN
set_domain_supply_net PD_SODIUM \
 -primary_power_net VN1 -primary_ground_net GN
```

Please refer this following document to get basic UPF example:
UPF Script Examples

Basic Command for Power Planning

Cadance Command

1. Need the logical connection:
   globalNetConnect - The globalNetConnect(CUI: connect_global_net) command is utilized to specify the logical power and ground connections, unless these are separately defined in the CPF (Common Power Format) or UPF (Unified Power Format). This command establishes the power nets for connecting power pins and assists in identifying which nets to use for tie-high or tie-low connections.
   Ex: globalNetConnect VDD_GCORE -type pgpin -pin VDD -inst *; Here Power net name is VDD_GCORE it will connect with the VDD pin of all instances.

2. Then, we need to set the track for the routing of the metal layer:
   This track file can be found in the PDK directory, which needs to be sourced in the flow. Additionally, it may need to be modified according to the project requirements.

3. We need to set different modes for the sroute and stripe:
   setAddStripeMode – this command is needed to modify the stripe creation mode.
   setSrouteMode – this command is needed to modify the Sroute creation mode.
   Ex: setAddStripeMode -skip_via_on_wire_shape {} -spacing_type center_to_center -stapling_nets_style side_to_side
   setSrouteMode -connectBrokenCorePin 1

4. Sroute creation:
   sroute – this command will create the power rail. We need to identify the layer containing the VDD and VSS pins and use that layer to make the power rail. In most cases, it will use the lowest metal layer.
   Ex- sroute -corePinLayer M0 -nets "VDD VSS" -corePinWidth $M0_width -connect corePin -corePinTarget none -allowJogging 0 -allowLayerChange 0 -area "$bound_llx $bound_lly $bound_urx $bound_ury"; here area will be the chip size

5. We need to set the via generation mode:
   setViaGenMode – this command will set the mode for via generation.
   Ex: setViaGenMode -viarule_preference $VIA0_master -allow_wire_shape_change false -optimize_cross_via true

6. For generating stripe:
   First of all, we need to set the stripe mode separately for each metal layer.
   Ex: setAddStripeMode -stacked_via_top_layer M1 -stacked_via_bottom_layer M0 -skip_via_on_wire_shape {} -skip_via_on_pin {} -spacing_type center_to_center -stapling_nets_style side_to_side -remove_floating_stapling false
   setAddStripeMode -trim_stripe core_boundary

for creating stripe, we need to use following command:
Ex: addStripe -layer M1 -direction vertical -nets "{$net_1st}" -width $M1_width -set_to_set_distance [expr $M1_step] -spacing [expr $M1_pitch] -start_offset [expr $bound_offset] -area "{$bound_llx $bound_lly $bound_urx $bound_ury}" -stapling "0.059 $m1_loc_y1 $M0_step:1" -snap_wire_center_to_grid Grid

7. Assign PG coloring:
colorizePowerMesh - Use this command to colorize (assign mask information) power structures, including power wires and power via metals, in multi-pattern layers. This command also fixes color violations of power metals including via metals.

Synopsys Command

1. First of all, need to create the PG port, PG net and connect power nets:
create_port – this command is needed to create the PG port.
Ex: create_port -direction inout -port_type power $pwr_nets
create_net – this command is needed to create the PG net.
Ex: create_net -power $pwr_nets
connect_pg_net - This command connects power, ground, and tie-off pins to the power and ground nets. It offers both automatic and manual modes, with automatic mode being the default. In automatic mode, the tool derives the power and ground nets from the design's power intent, requiring committed UPF descriptions and logic libraries with power and ground pins. In manual mode, you specify the power and ground nets.
Ex: connect_pg_net -automatic

2. We need to create the routing track for metal layers:
This track creation file can be found in PDK directory and sometimes we need to modify it as per project requirement.

3. Some app option is needed to properly create the power rail and stripe:
set_app_option – This command will set the app options.
Ex: set_app_options -name plan.pgroute.siterow_name -value unitW48M143H117
set_app_options -name plan.pgroute.stop_rail_stdcell_pin -value true

4. For creating the power and ground rail:
The lowest matel layer will be used for creating rail.
create_pg_std_cell_conn_pattern – this command is creating a standard cell rail connection pattern.
Ex: create_pg_std_cell_conn_pattern M0_mesh -rail_width $M0_width -layers M0 -rail_shift $M0_track_offset
set_pg_strategy – The pattern which is created using create_pg_std_cell_conn_pattern can be instantiated in a design by set_pg_strategy command to create standard cell rails.
Ex: set_pg_strategy M0_strategy -core -pattern "{name: M0_mesh} {nets: $net_1st $net_2nd}"
compile_pg – it will create power and ground rail based on the specified power ground startegies.
Ex: compile_pg -strategies {M0_strategy}

5. For creating power and ground stripe:
   create_pg_wire_pattern - Creates a power ground wire pattern.
   Ex: create_pg_wire_pattern M1_mesh_1st -direction vertical -layer M1 -width $M1_width -pitch "$M1_step $M0_step" -center $M1_loc_x_1st -high_end_reference_point [expr $M1_loc_y_1st + $M1_length/2 + 0.008] -low_end_reference_point [expr $M1_loc_y_1st - $M1_length/2 + 0.008] -spacing [expr $M1_pitch*1] -track_alignment track
   set_pg_strategy – The pattern which is created using create_pg_wire_pattern can be instantiated in a design by set_pg_strategy command to create power ground mesh.
   Ex: set_pg_strategy M1_strategy_1st -core -pattern "{name: M1_mesh_1st} {nets: $net_1st}"
   set_pg_via_master_rule - Creates a new via master rule for the power and ground network. The via rule is defined by the contact code, location of contact code inside the intersection box, cut spacing, multiplication, and so on.
   Ex: set_pg_via_master_rule VIA01_rule_1st -contact_code $VIA0_master -via_array_dimension {1 1} -cut_spacing {1 1}
   set_pg_strategy_via_rule - Specifies the via rule between strategies and collections of existing shapes.
   Ex: set_pg_strategy_via_rule VIA01_strategy_1st -via_rule "{{intersection: adjacent} {via_master: VIA01_rule_1st}}"
   Compile_pg - Creates a power and ground network including straps, rings, macro connections, standard cell connections and via connections based on the specified power ground strategies.
   Ex: compile_pg -strategies {M1_strategy_1st} -via_rule {VIA01_strategy_1st}

6. Assign PG coloring:
   derive_pg_mask_constraint - This command will colorize the PG layer that has a mask pattern.
   Ex: derive_pg_mask_constraint -derive_cut_mask -always_align_color_layers {M0 M2} -overwrite

Verify PG Network

Cadance Command

- **verify_drc** – This command supports all the tech nodes. Use the Verify DRC form through Verify - Verify DRC to check the DRC violations.
- **verifyPowerVia** - This command provides a variety of power-rail overlap checks to look for missing power-grid vias.

Synopsys Command

- **check_pg_missing_vias** – Checks missing vias within the power and ground network.
- **check_pg_connectivity** – Verifies physical connectivity of the power ground network.
- **check_pg_drc** - Verifies and reports technology routing design rules violations and illegal overlaps of PG net objects.

References

1. https://www.youtube.com/watch?v=X6e3Yshs0Ys
2. https://www.physicaldesign4u.com/2020/01/power-planning.html
3. https://vlsitalks.com/physical-design/powerplan/
4. https://chipedge.com/power-planning-in-vlsi-design-balancing-efficiency-and-performance/
5. https://www.vlsi4freshers.com/2020/01/power-planning.html
6. https://siliconvlsi.com/what-is-an-always-on-cell-in-vlsi/
7. https://www.youtube.com/watch?v=hqibU4UqoLk
8. https://ivlsi.com/aon-cell-vlsi-physical-design/
9. https://www.synopsys.com/glossary/what-is-low-power-design.html
10. https://www.maven-silicon.com/blog/what-is-low-power-design-in-vlsi/
11. Implementing Low Power Design Through Voltage Sca https://semiengineering.com/defining-power-intent/ling in VLSI | System Analysis Blog | Cadence
12. https://semiengineering.com/defining-power-intent/
13. Innovus User Guide -- CPF 1.1 Script Example
14. When and why is the globalNetConnect / connect_global_net command run?
15. https://www.scribd.com/presentation/530820057/CPF-and-UPF