Low Power Digital Design Fundamental
– An EDA Perspective

Richard Chou (rchou@cadence.com)

Director of SW Engineering

Cadence Design System, Inc.

March, 2018

cadence®

About this presentation

- Focus on electronic design automation(EDA), not circuit design
- Focus on low-power digital IC design, NOT analog
  - ASIC designs based on standard-cell methodology, not FPGA
- Focus on physical implementation, NOT device physics
- Focus on industrial experience, NOT theoretical ideas
- Prepare you to be an engineer, NOT a professor

Agenda

- Introduction
- Low Power Physical Implementation
- Advanced Low Power Techniques
- Devices – Now & Future
- Summary

Low Power Design Issues Impact Profitability

Different drivers in different verticals

Mobile/Hand-held

Battery Life
Unit Cost (chip package)

Consumer/Digital Home

Unit Cost (chip package)
Unit Cost (fans etc.)
Reliability

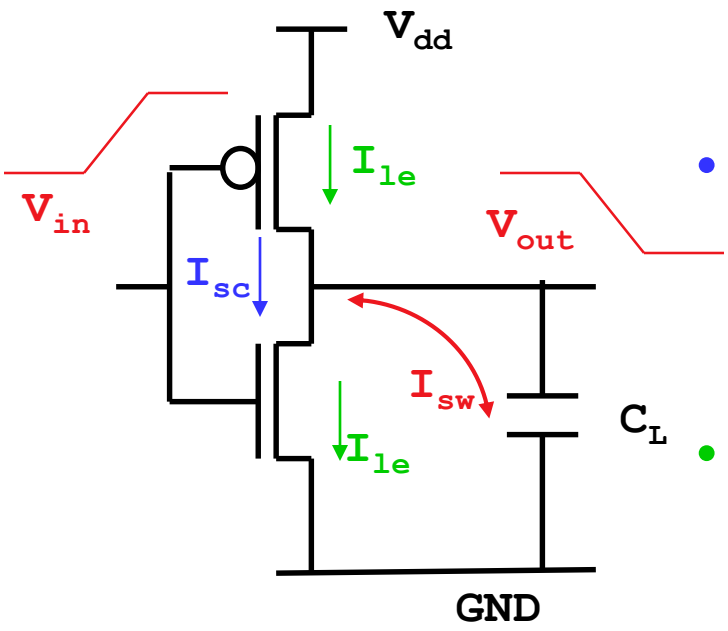
Network/Data Center

Power Efficiency
Total Cost of Ownership
Reliability
Green

Low power requirements drive different design decisions:

- Product design architecture and integration decisions
- IP make versus reuse versus buy decisions
- Manufacturing process decisions

Device Current Components

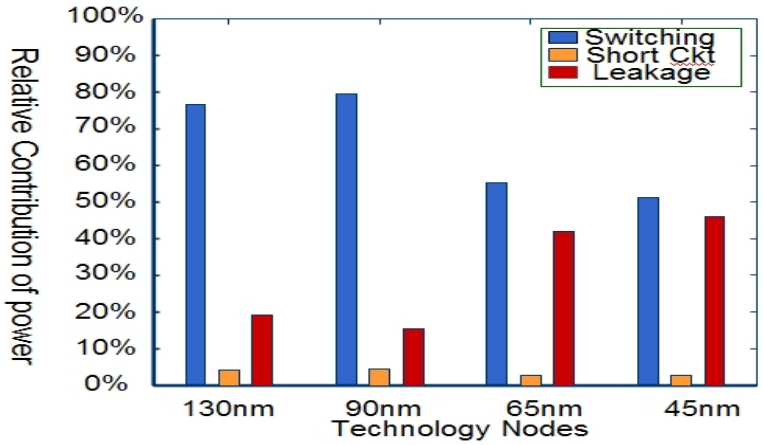


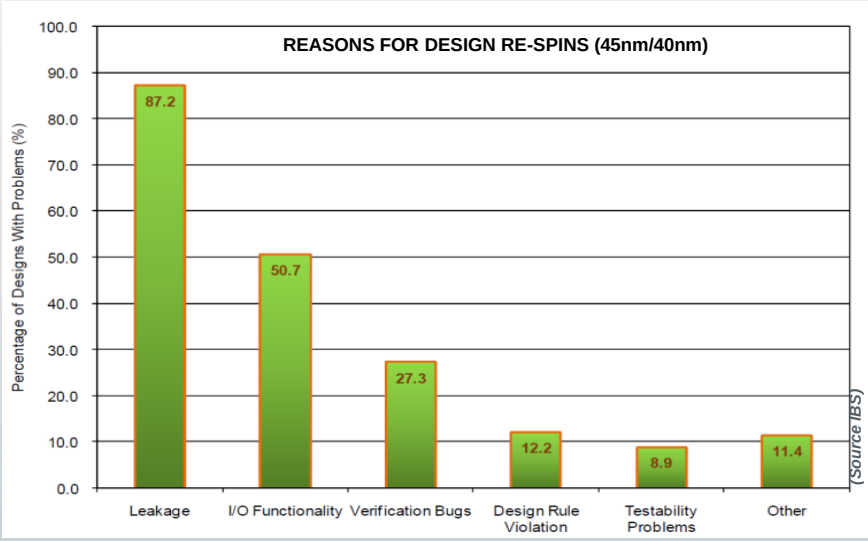
- **Dynamic Switching Power**
  - Due to charge/discharge of load cap
  - $I_{sw} \sim C_L V_{dd}^2$
- **Dynamic Short-circuit Power**
  - Due to direct current path from Vdd to ground during output switching
  - $I_{sc} \sim$ input_slew / C_L
- **Static Leakage Power**
  - Due to subthreshold & gate leakage
  - $I_{le} \sim K*e^{qV_{gs}/kT}(1-e^{-qV_{ds}/KT})$

$$P_{total} = C_L.V_{DD}^2.f_{clk}.a_{0\rightarrow1} + V_{DD}.I_{short\text{-}circuit} + V_{DD}.I_{leakage}$$

Types of Power Consumption

- Dynamic (switching) power consumption
- Short circuit power consumption
- Static (leakage) power consumption





Biggest reason for design failure: Leakage

cadence

Low-Power Solution (Cadence)

System Level

- Stratus™ High-Level Synthesis (HLS)
- Palladium® Dynamic Power Analysis (DPA)
- Chip-package co-design with Sigrity™ and Voltus™ solutions

Func. Verification

- Xcelium® Simulator
- Palladium emulator
- JasperGold® Formal Power App
- Analog Mixed-Signal Designer

Synthesis and DFT

- Genus™ Logic Synthesis
- Modus DFT and ATPG
- Conformal Low Power
- Joules RTL Power Estimation

P&R, MS & Signoff

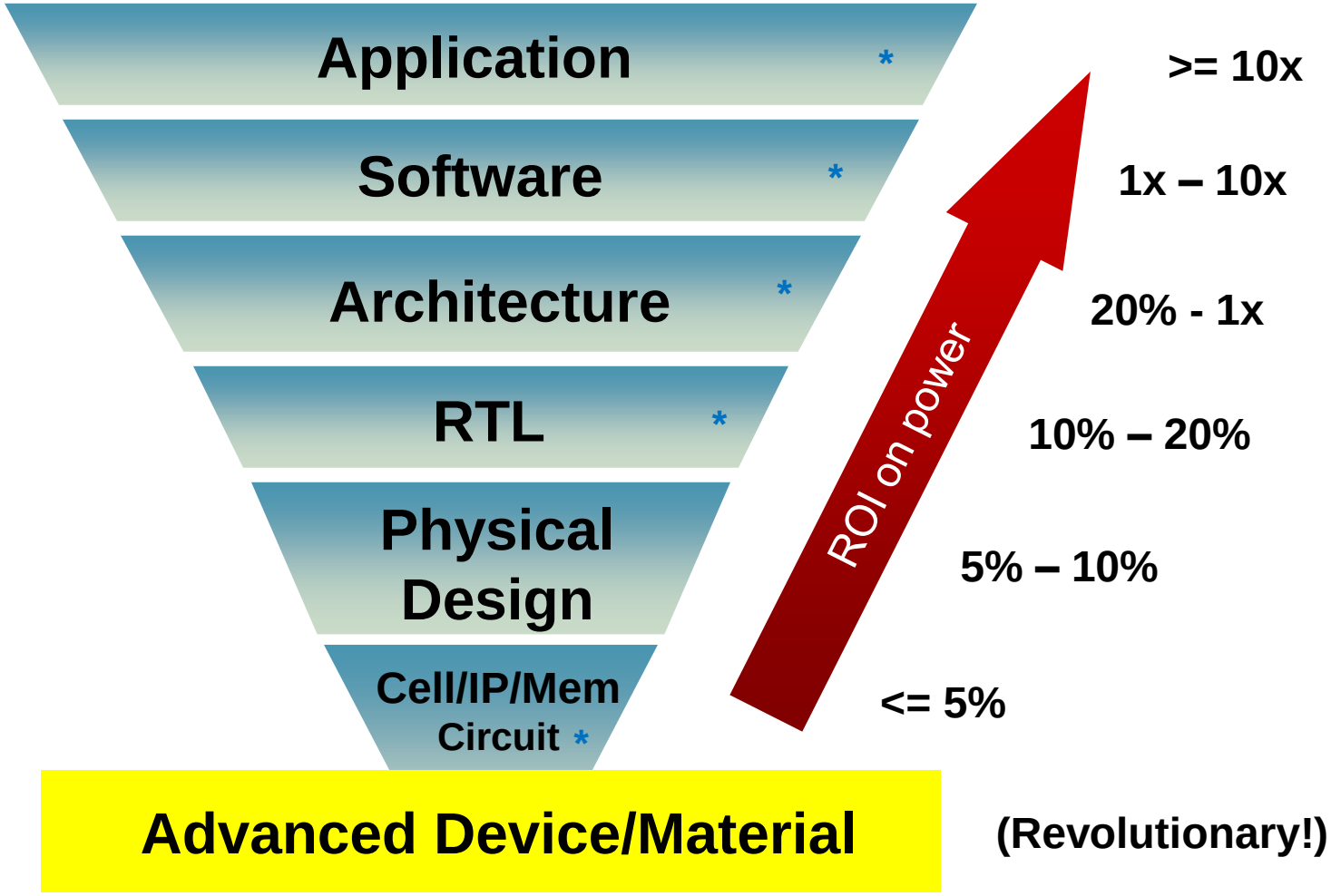
- **Innovus™ Implementation System**
- OA-based Mixed Signal with Virtuoso® technology
- MS static checks with Conformal® LP
- Tempus ECO
- Voltus IC Power Integrity Solution

IP

- Energy-efficient Xtensa® cores
- LPDDR, PCI Express® (PCIe®), Ethernet, MIPI, USB, eMMC
- Analog mixed-signal IP including ADC/DAC, AFE, SerDes, PVT Monitors, and power management IP

- **Aspects to consider in low power solution:**
  - **Estimation & Simulation**
  - **Synthesis & Implementation**
  - **Verification & SignOff**

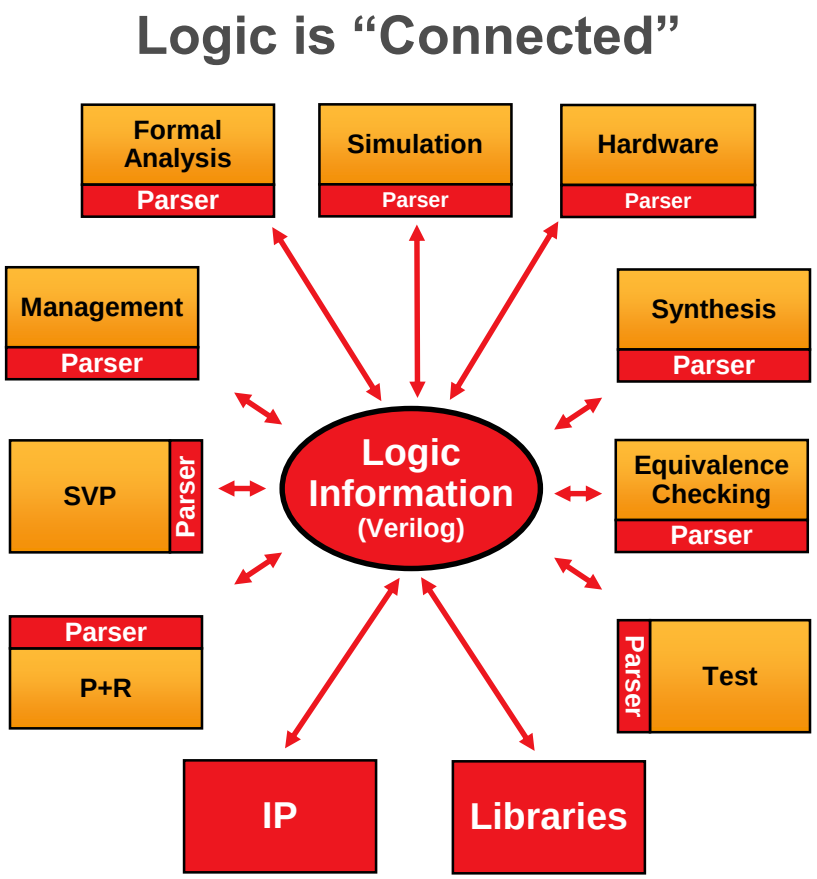
ROI on Power Optimization at Various Levels

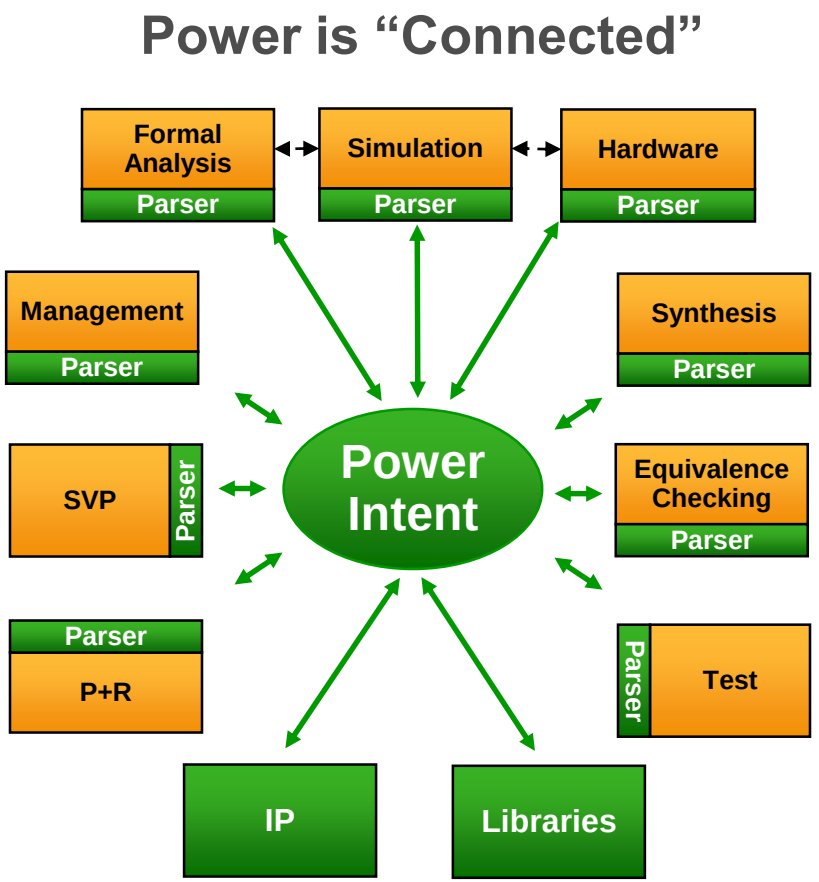


(* See Appendix for the "Global Picture" on reducing power)

Low Power Physical Implementation

The Need for Power Intent Information (Spec)

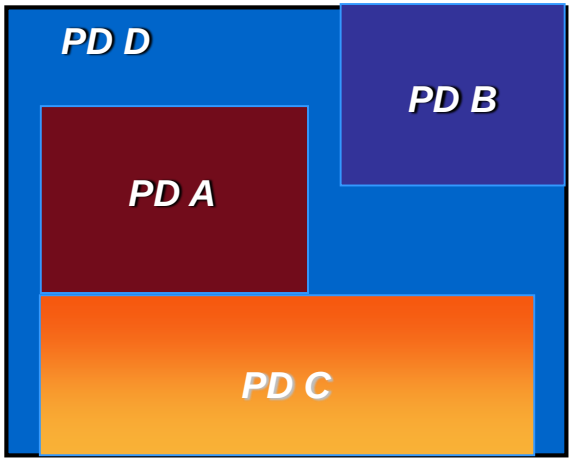




cadence

Power Intent - 2 Industry Formats

- **Common Power Format (CPF)**
- **Unified Power Format (UPF – aka IEEE1801)**
  - IEEE1801 standard version
- **Content of Power Intent File**
  - Power domain & membership
  - Power domain operating modes (power modes, power state table, port state)
  - Power domain interface rules/strategies
  - Power supplies, voltages, connection, and association with power domains

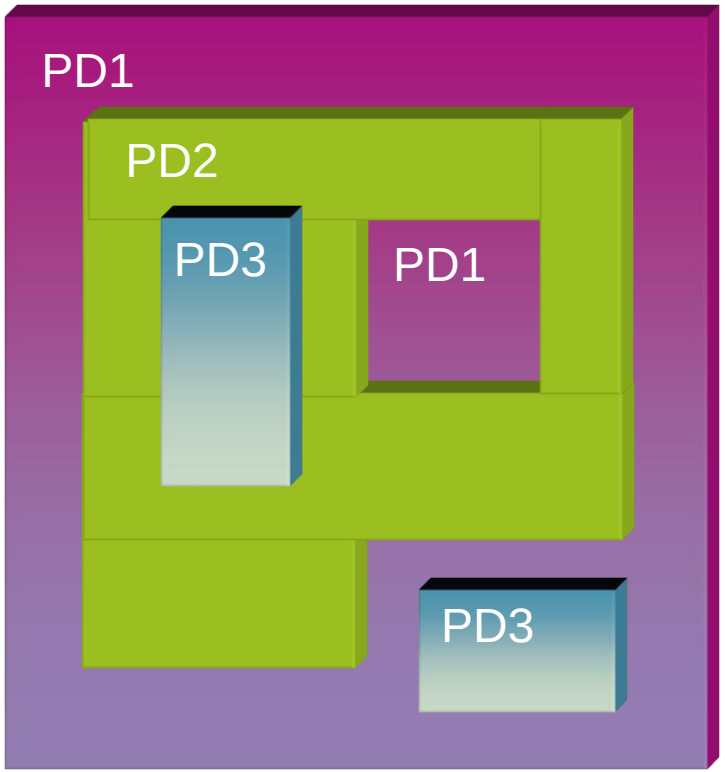


	PD A	PD B	PD C	PD D
PM1	1.2v	1.2v	1.2v	1.2v
PM2	0.8v	off	1.2v	1.2v
PM3	0.8v	off	off	off
PM4	0.8v	1.2v	1.2v	1.2v

Power Domain

- **Possible definitions:**
  - Based on power net grouping (more physical oriented)
  - Based on power “characteristics” groupings (more logical oriented)

- Physical-oriented power domain types:
  - **Default power domain (PD1)**
    - Non-default power domain in the middle
    - Default power domain in the middle
  - **Donut shape power domain (PD2)**
  - **Nested power domain (PD3)**
    - Physical hierarchy vs. logical hierarchy
  - **Disjoint power domain (PD3)**



SoC Power Domain-based Power Consumption

$$P_{total} = C_L . V_{DD}^2 . f_{clk} . a_{0 \rightarrow 1} + V_{DD} . I_{short\text{-}circuit} + V_{DD} . I_{leakage}$$

$$P_{SoC} = (C_1 V_{DD\_1}^2 f_1 + V_{DD\_1} I_1) + \quad \text{// power domain 1}$$

$$(C_2 V_{DD\_2}^2 f_2 + V_{DD\_2} I_2) + \quad \text{// power domain 2}$$

$$(C_3 V_{DD\_3}^2 f_3 + V_{DD\_3} I_3) + \quad \text{// power domain 3}$$

$$\ldots$$

General Low Power Techniques (Special Cells)

- **Multiple Supply Voltages (MSV)** (aka, MSMV, MV)
  - Level-shifter cell
  - Always-on cell
- **Power Shut-Off (PSO)**
  - Power switch cell (aka: power gate, mtcmos cell)
  - Isolation cell
  - Always-on cell
  - Combo cell, enabled level-shifter cell (ELS)
  - State-retention flip-flop (SRFF)
- **Clock Gating**
  - Integrated clock-gating cell (ICG)
- **Multi-Vt Cells (MT-CMOS)**
  - High-Vt (HVT) : slower, but low leakage
  - Standard-Vt (SVT)
  - Low-Vt (LVT) : faster, but higher leakage

General Low Power Techniques (Design Flow)

- **Floorplanning**
  - Power domain fence, shape, location definition
- **Placement**
  - Power-aware placement (eg, shorter high-frequency/high-voltage/high-cap nets, …)
  - Power domain interface gate placement (eg, isolation and shifter)
- **Optimization / Buffering**
  - Power-aware buffer types (ao vs non-ao) across domains and in feed-through
- **Clock Tree Synthesis**
  - Clock gating
  - Utilize useful skew
  - Same power-aware as IPO buffering
- **Routing**
  - Power-aware routing (eg, shorter high-frequency/high-voltage/high-cap nets, …)
  - Domain-aware routing control
- **Leakage optimization**
  - Use of high-Vt / longer gate / stacking cells

Power Shut-Off & Power Switches

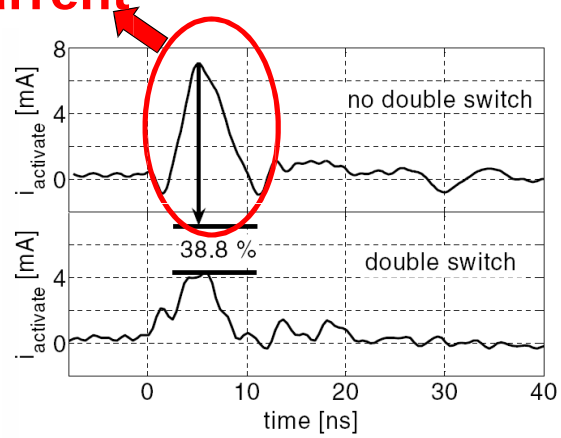
Power Switch (Power-Gating Cell) Considerations

- How many
  - Too few – IR (power dissipation)
  - Too many – Leakage, area overhead
- How to chain – rush current, routing resources
  - Simultaneously turn on all switches is a disaster
  - Power management unit controls the stage enable signals
  - Single in/out vs tree-style vs hybrid
- How to route power nets – column/row/grid style
  - Randomly place is NG
  - >= 3 power nets: VDD-AO, VDD-OFF, VSS
  - Make sure aligned in H/V direction
  - Choice affected by H/V routing resources availability

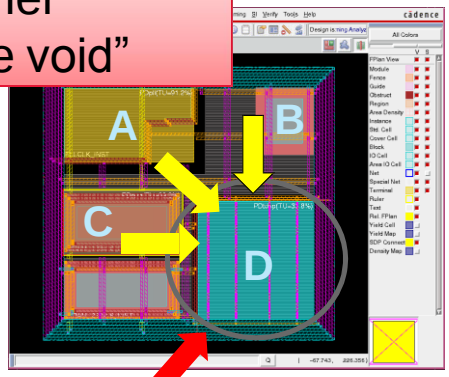
Rush current

- Faster is always better, right?
  - Usually yes, but not necessarily for **rush current**

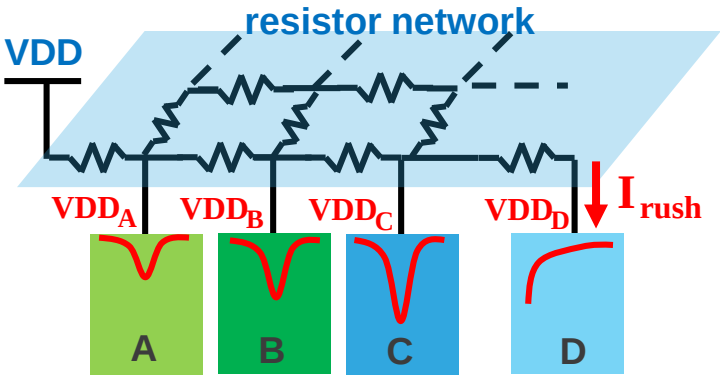
- What is rush current?
  - When a PSO domain turns on, current is drawn in to the PSO domain as the circuits start to become active.



Current from other domains "fill the void"

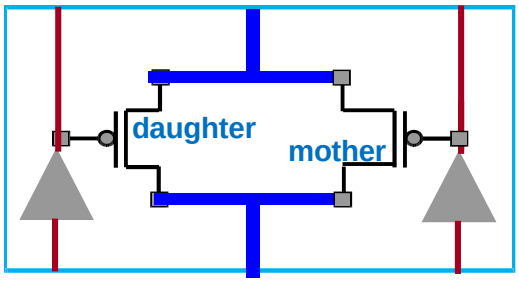


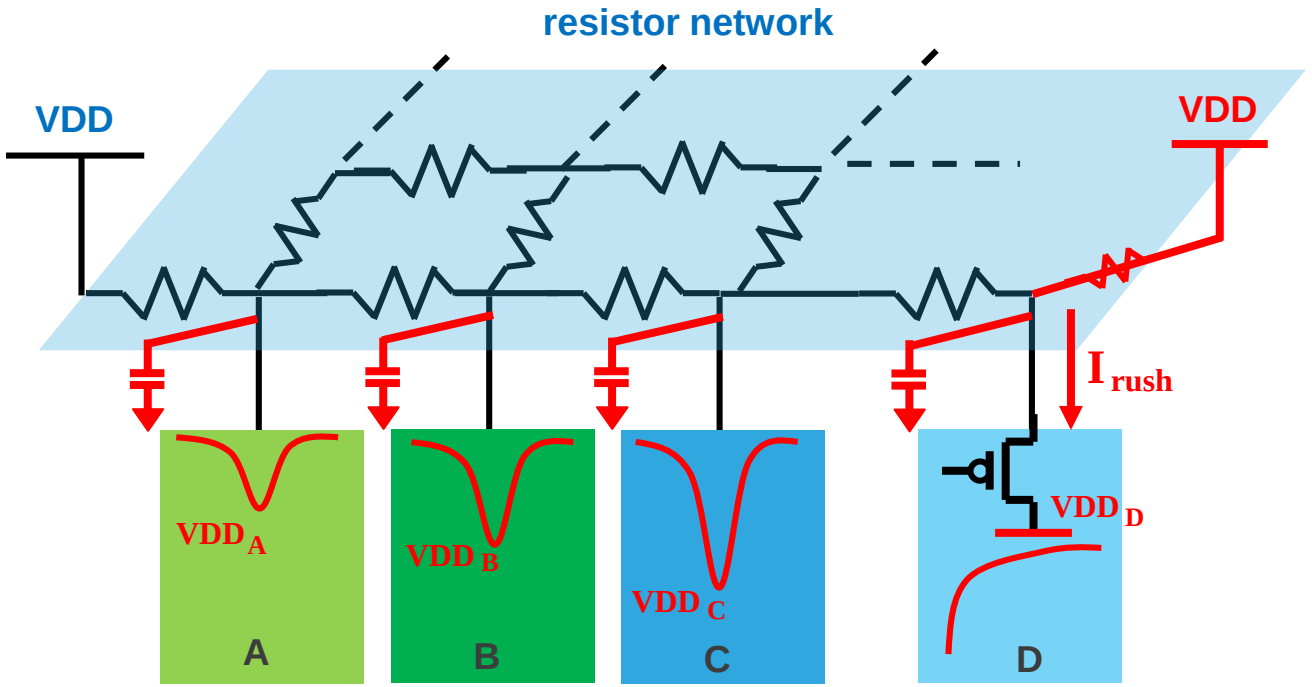
PSO domain "wakes up"

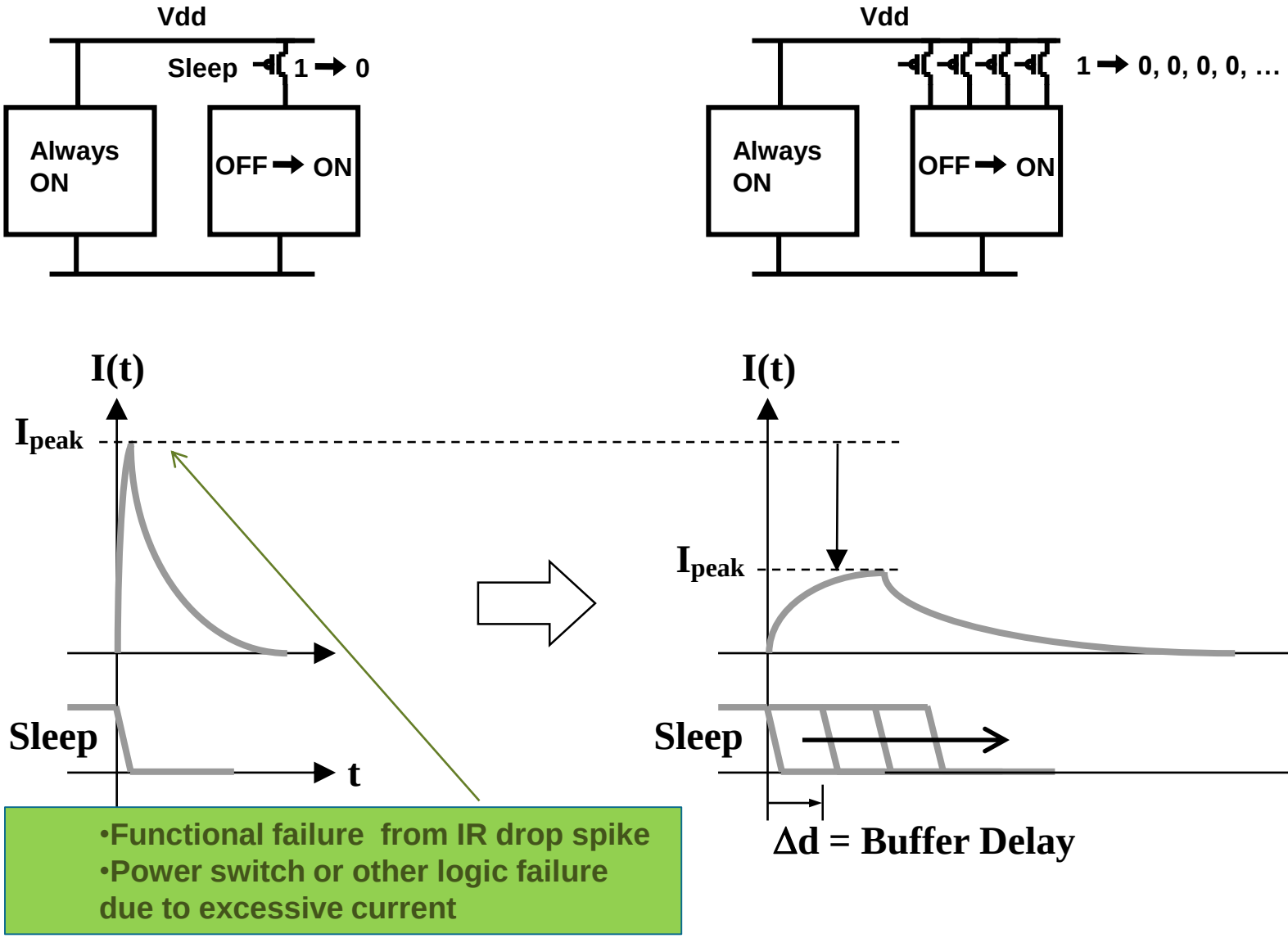


Rush current – How to Rectify

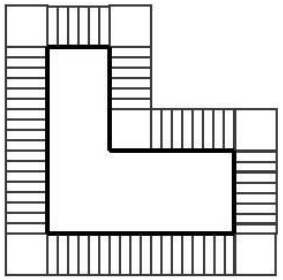
- 2-stage charging (switch enabling)
- Add decoupling (bypass) capacitors
- Move power domain closer to power source
- More robust power grid



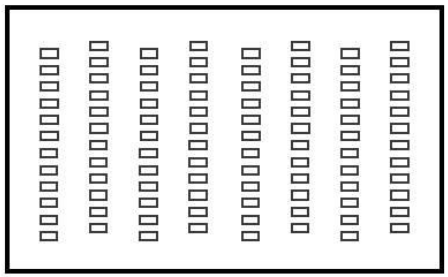




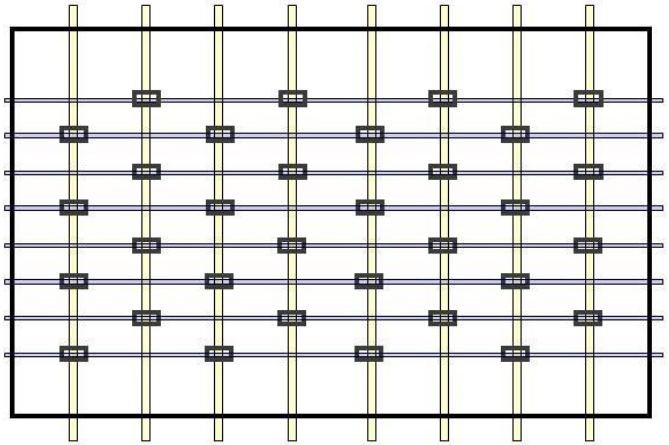
Power Switch Insertion and Placement



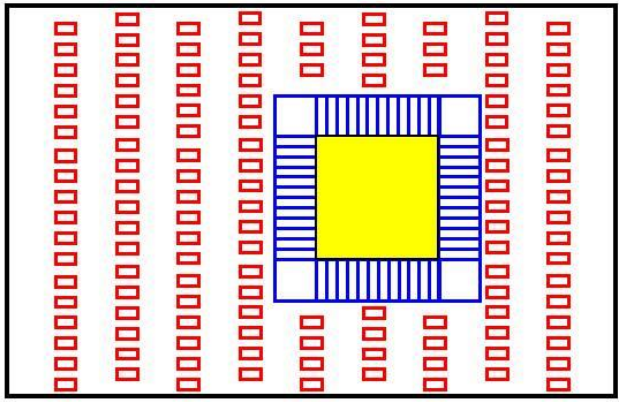
Ring



Checkerboard Column/Row

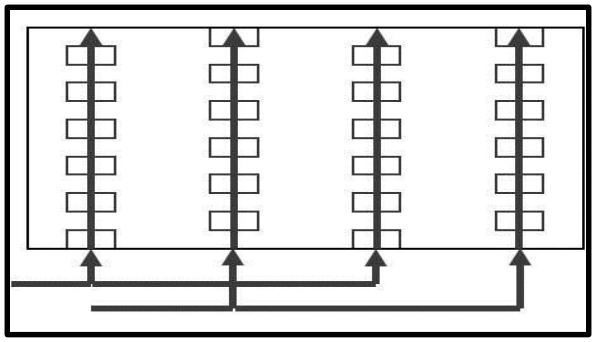


Under Power Grid

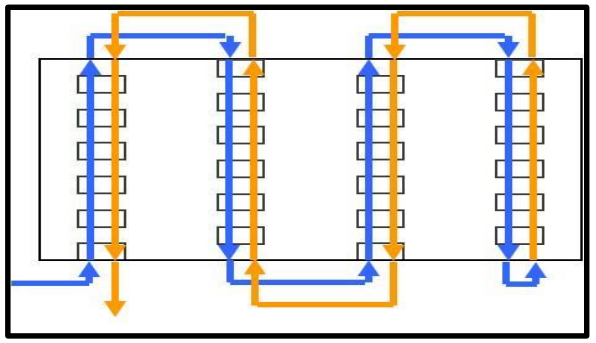


Mixed Styles

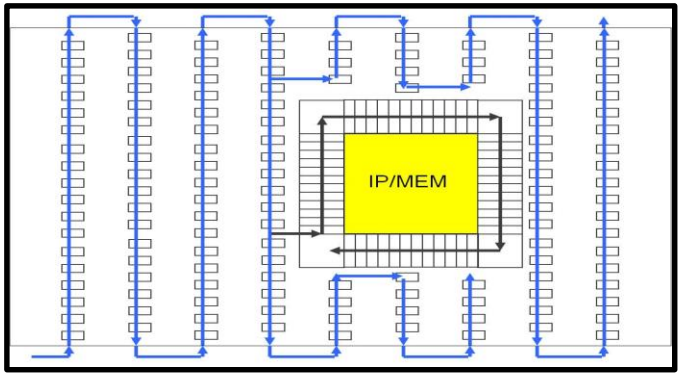
Flexible power switch enable chaining options



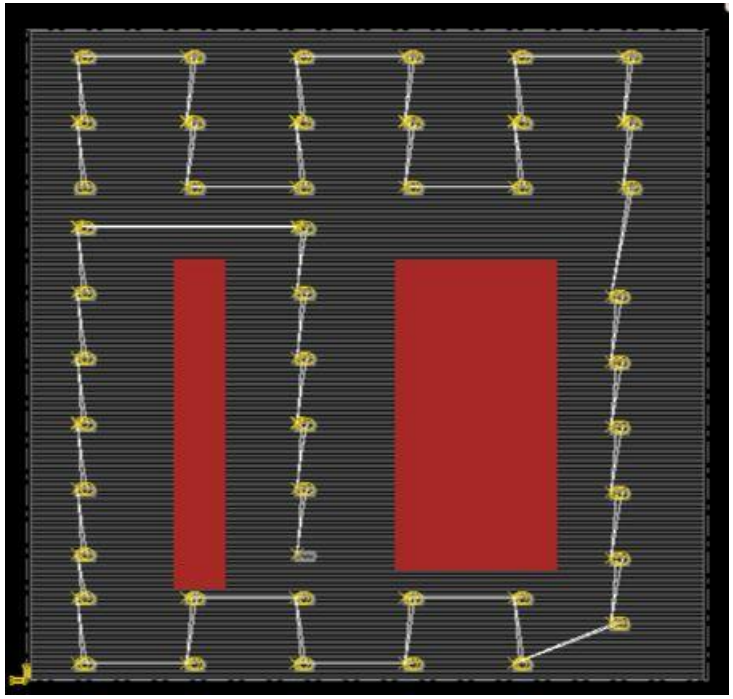
Staged Simultaneous Enable



Daisy Chain with Loopback (Ring and Column)

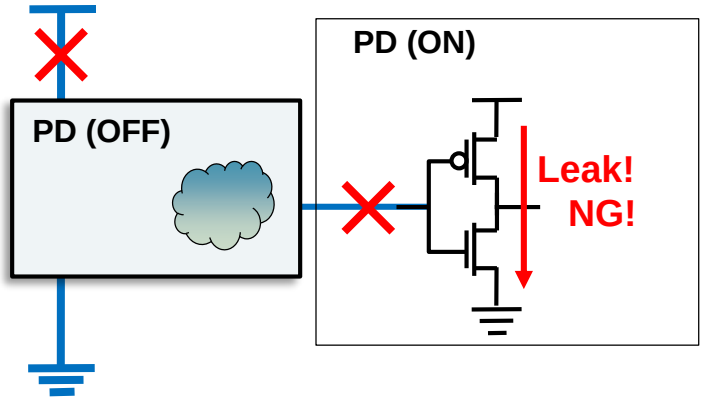


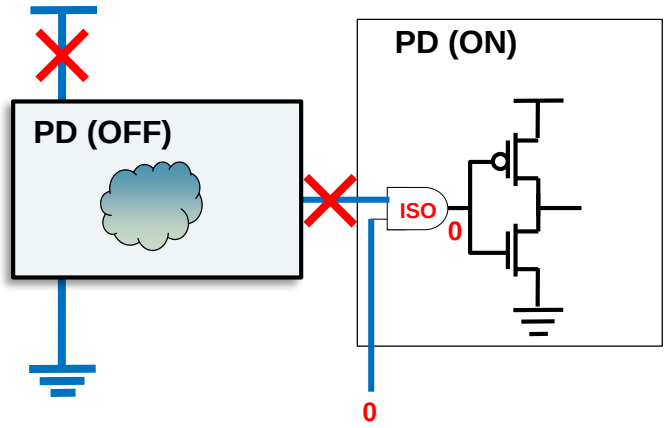
Single Input/Output Chain (TSP) - Example

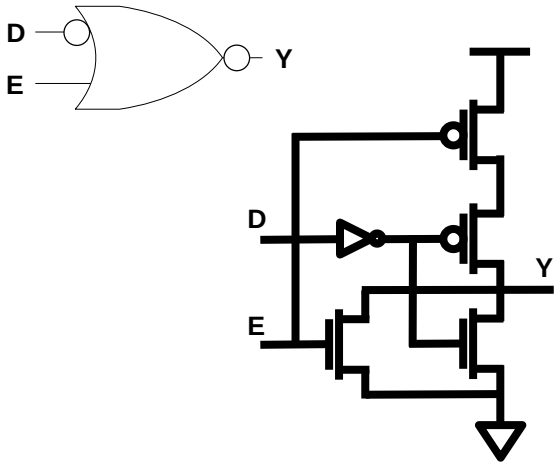


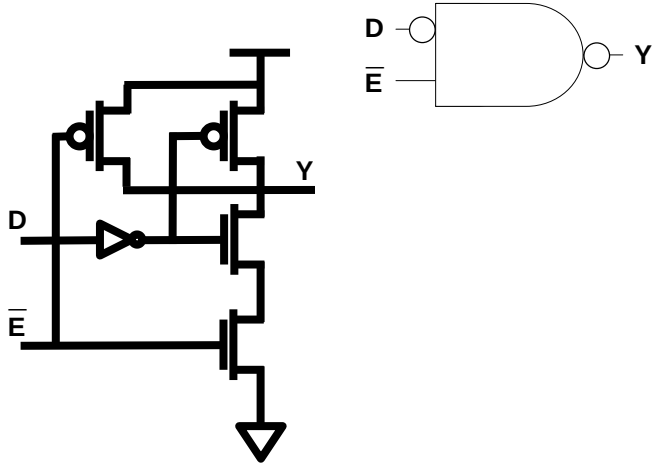
Power Shut-Off & Isolation

Isolation Function & Common Isolation Cells



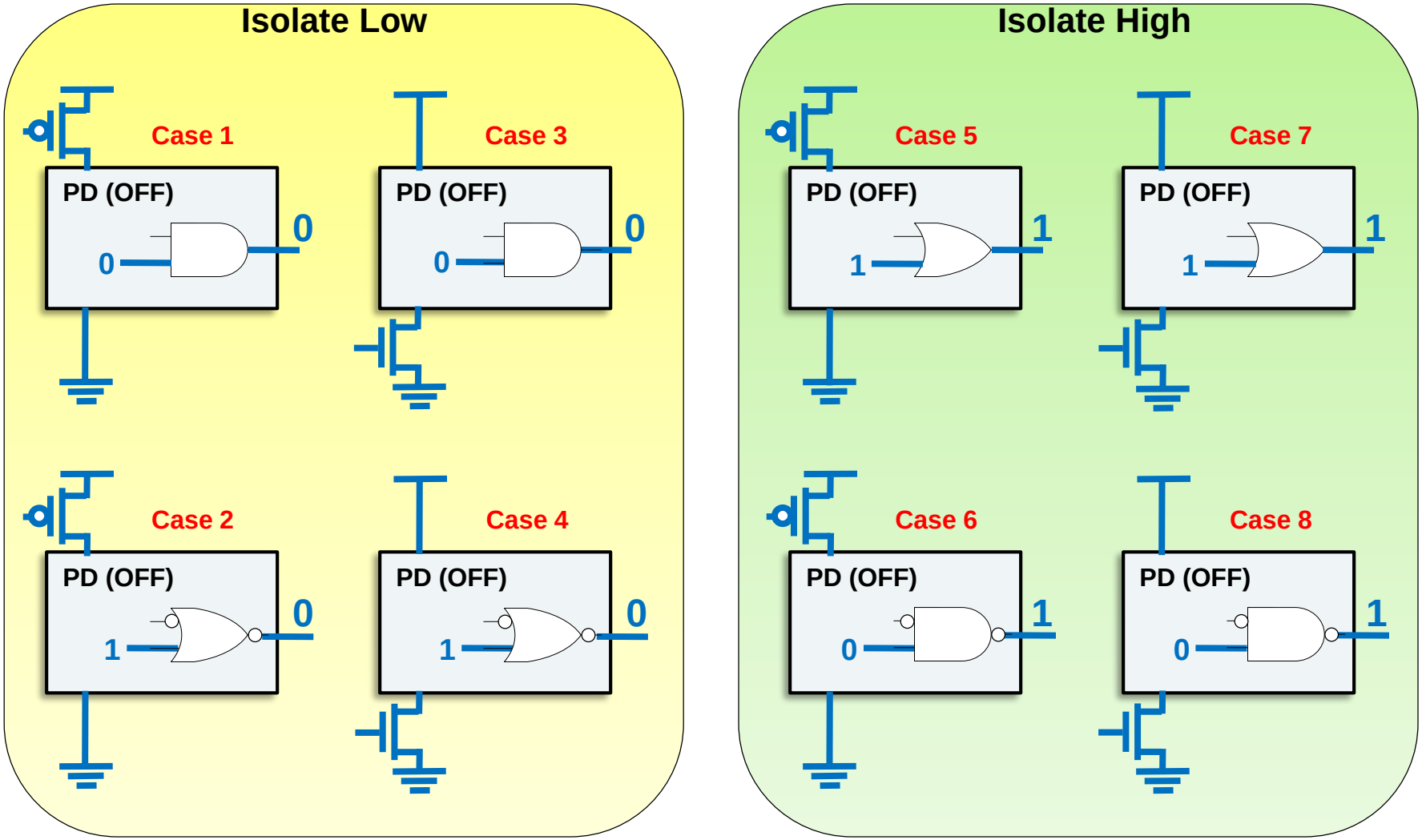






Isolation: which doesn't work?

Non-AO version, location = FROM domain

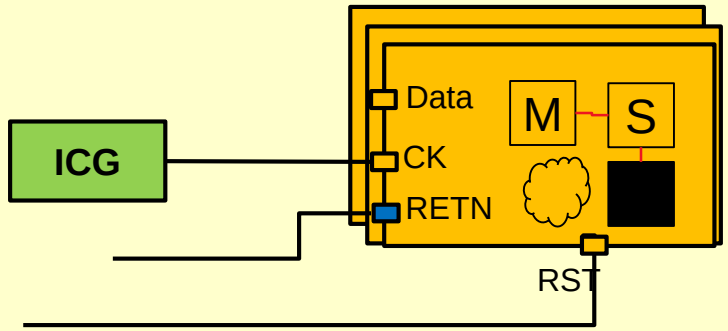


cadence

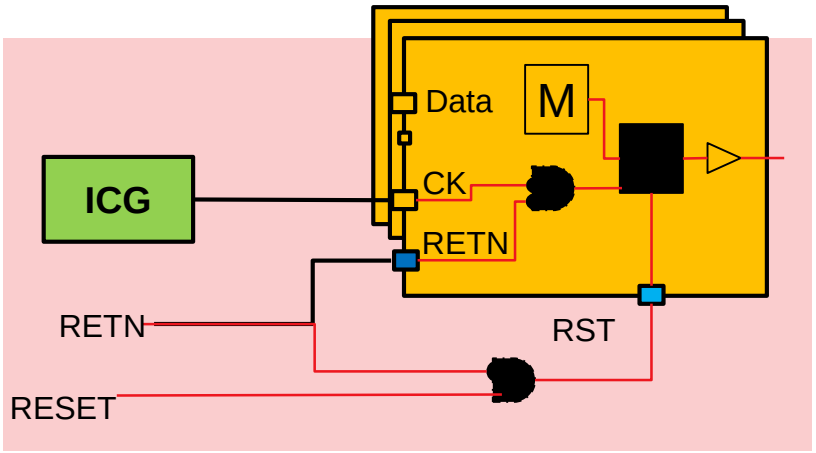
Power Shut-Off & State Retention

State-Retention Cell Styles

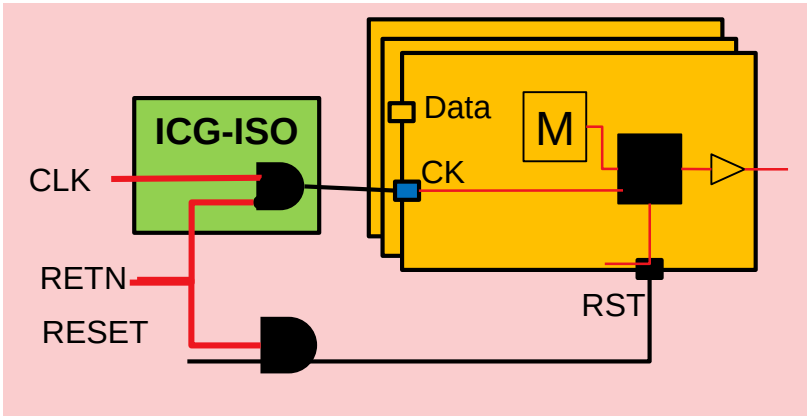
Balloon



General Live slave



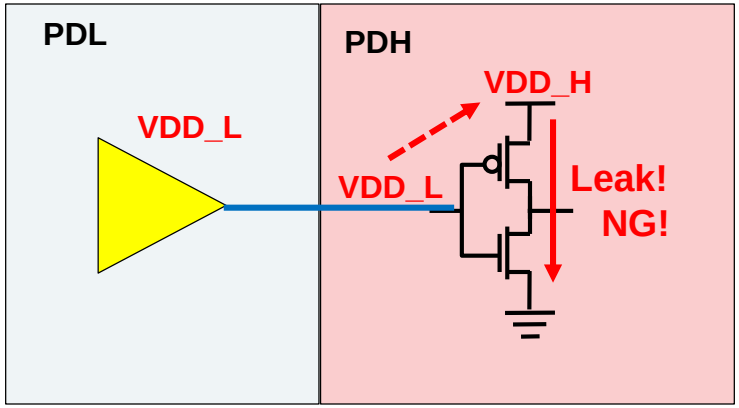
Zero Pin Live slave

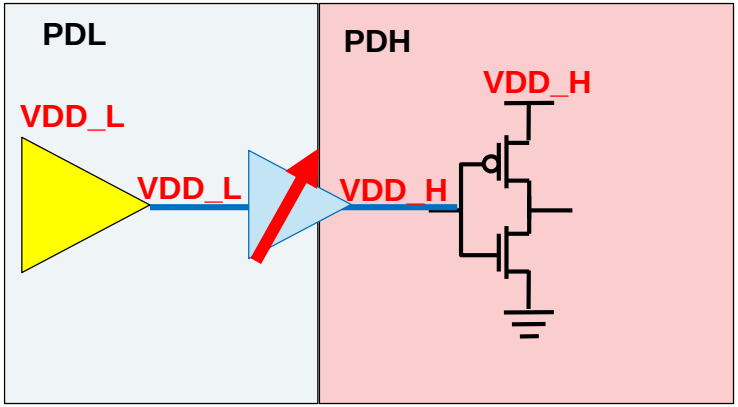


cadence

Multiple Voltages & Level Shifter

Level Shifter Function

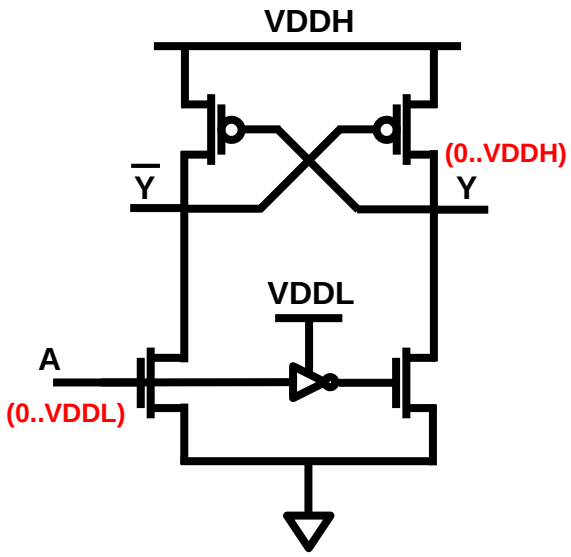




- Flavors
  - Pure shifter without enable
  - Shifter combined with isolation function
    - Enabled level shifter (ELS) / Combo cell

Common Level Shifter Circuits

Conventional Level Shifter

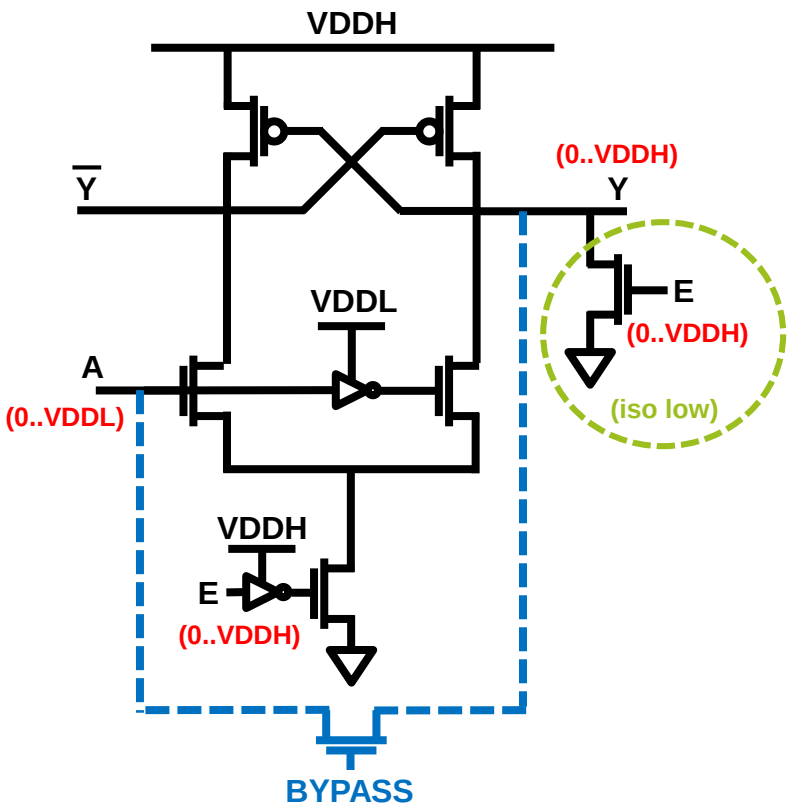


2 Flavors:

- Rail Pin == VDDH (valid loc == TO)
  2nd Power Pin == VDDL

- Rail Pin == VDDL (valid loc == FROM)
  2nd Power Pin == VDDH

Isolation + Level Shifter Combo With Enable

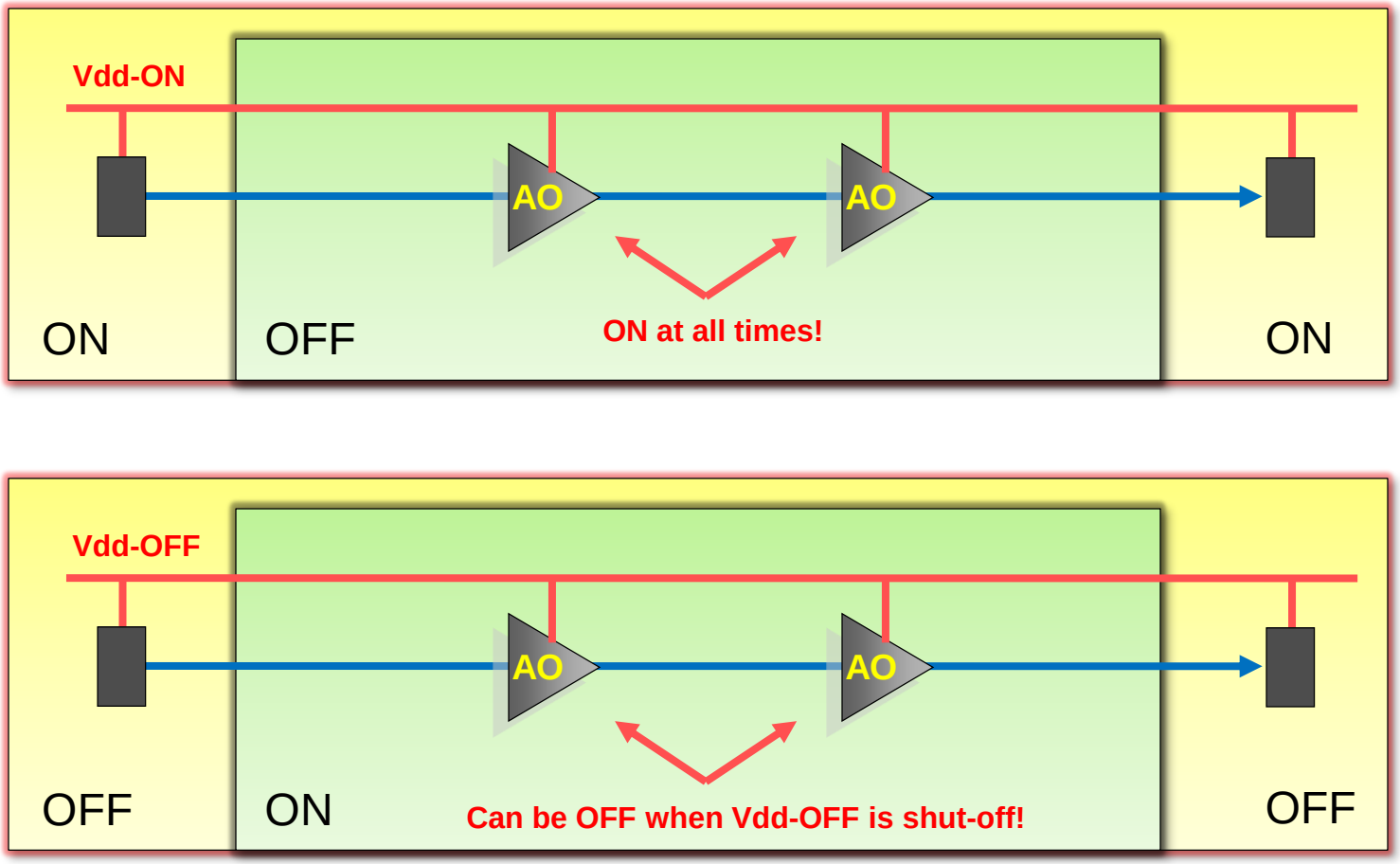


Rail Pin == VDDH (valid loc == TO)
2nd Power Pin == VDDL

NOTE: conceptual only! They are not the most efficient level shifter circuit!
Ways to reduce contention and improve delay are available in literature.

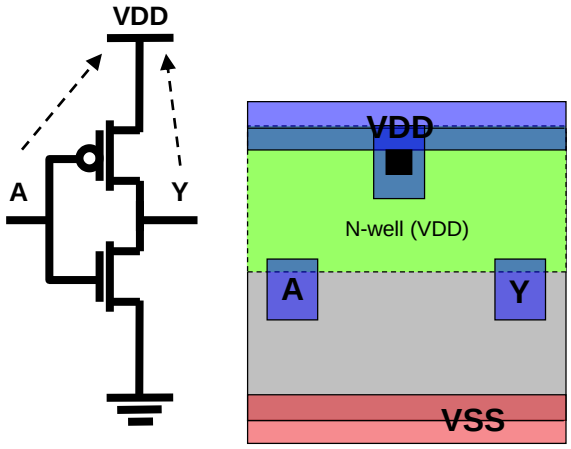
Multiple-Power-Domain Buffering

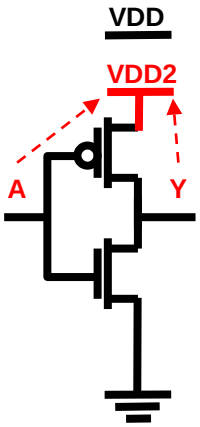
Domain Feed-Through Buffering

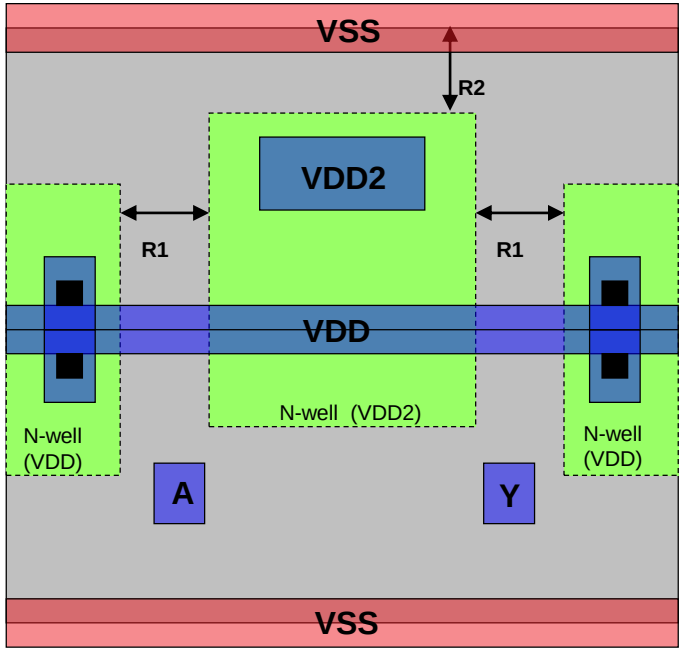


Always-On Buffer

- The actual supply is from the 2nd power pin, not power rail
  - **Can be placed anywhere, feed-through any domain** (aka. BYOP - bring your own power)
  - Larger than regular cell (normally 2x row)
  - Secondary power pin → Requires secondary power route

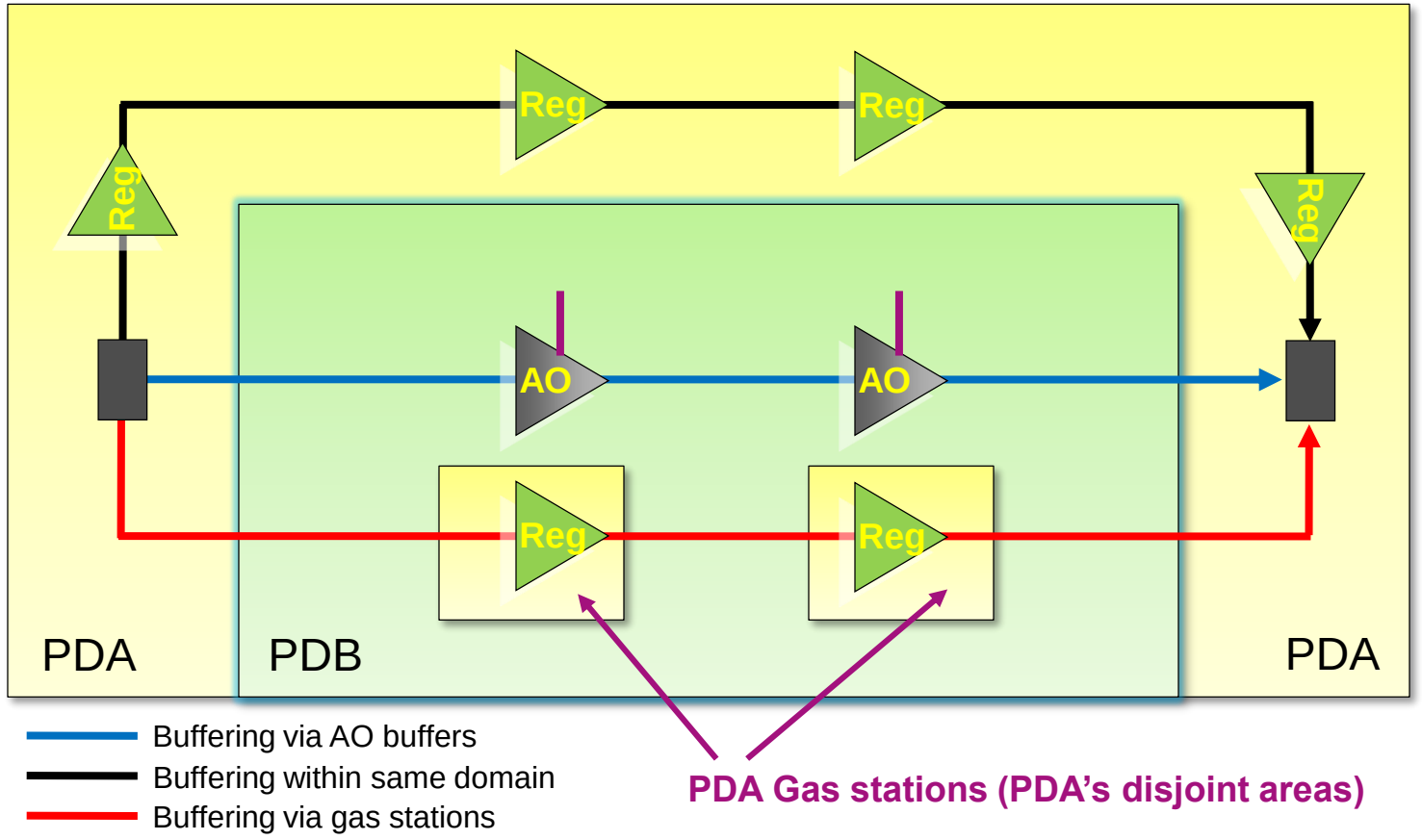






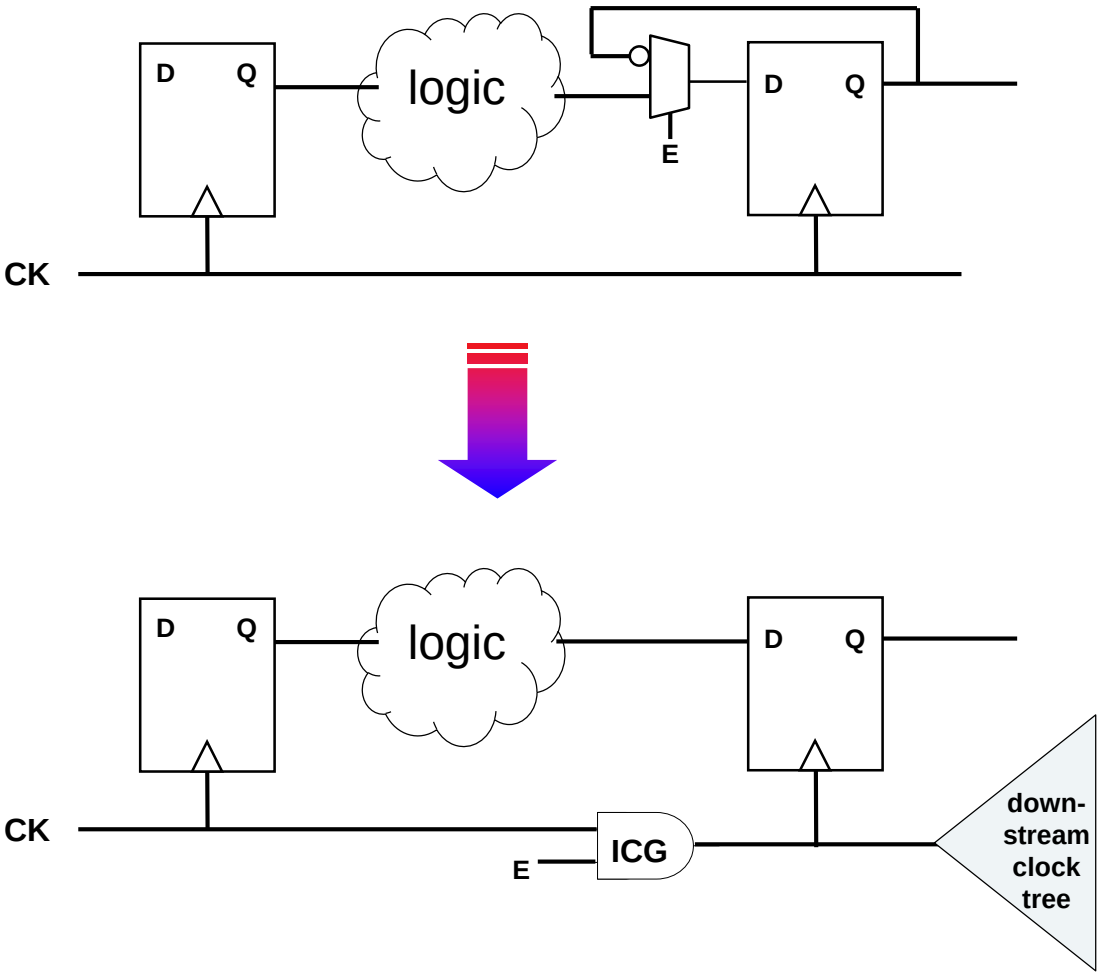
Gas Stations

- User created disjoint power domain areas(islands) in floorplan to allow regular buffers to “hop” through



Clock Gating

Traditional vs Clock Gating



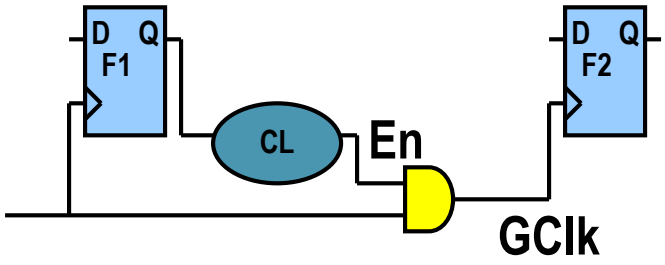
- Pro:
  - No race condition
  - Simple to analyze
- Con:
  - Bigger area
  - Bigger power
    - (one per FF)

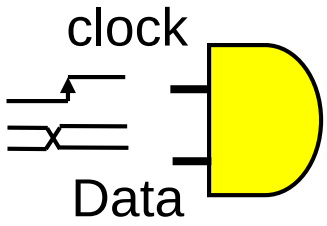
- Pro:
  - **Lower power**
  - Smaller area
    - (shared by many)
- Con:
  - If E is late:
    - clock glitch
    - clipped clock

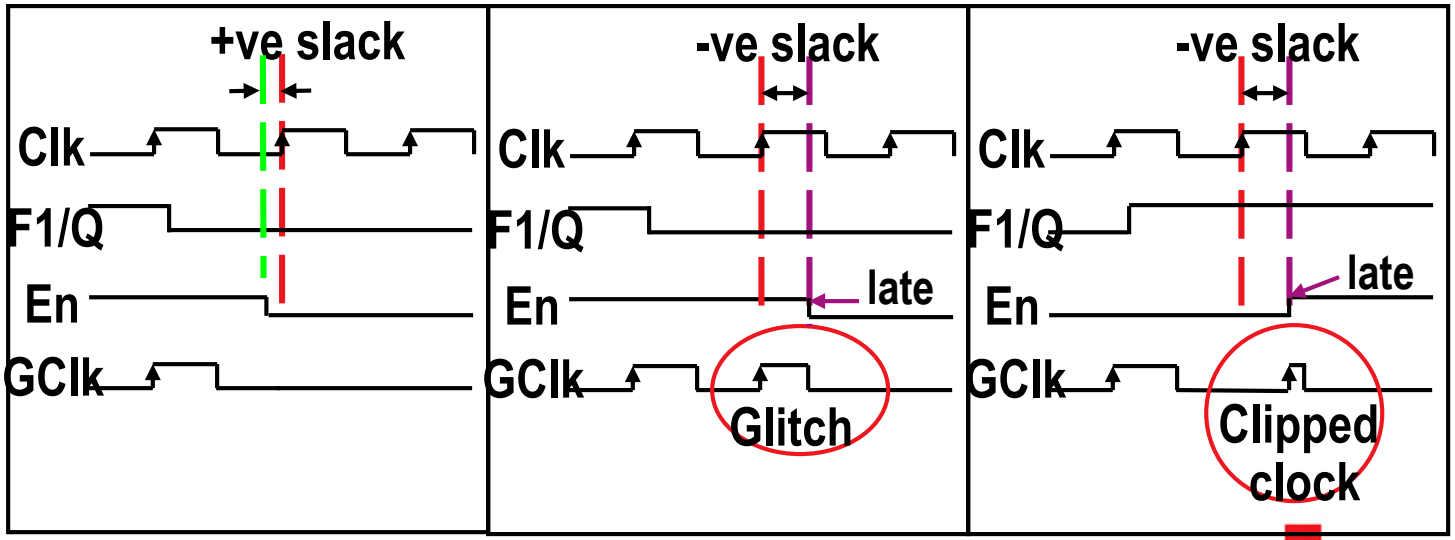
cadence™

What is Clock Gating Setup Violation ?

- Clock gating setup check : to ensure the controlling data signals are stable before the clock becomes active. The arrival time of the leading edge of the clock signal is checked against both edges of any data signal feeding the data pins to prevent at glitch at the leading edge of the clock pulse or a clipped clock pulse.

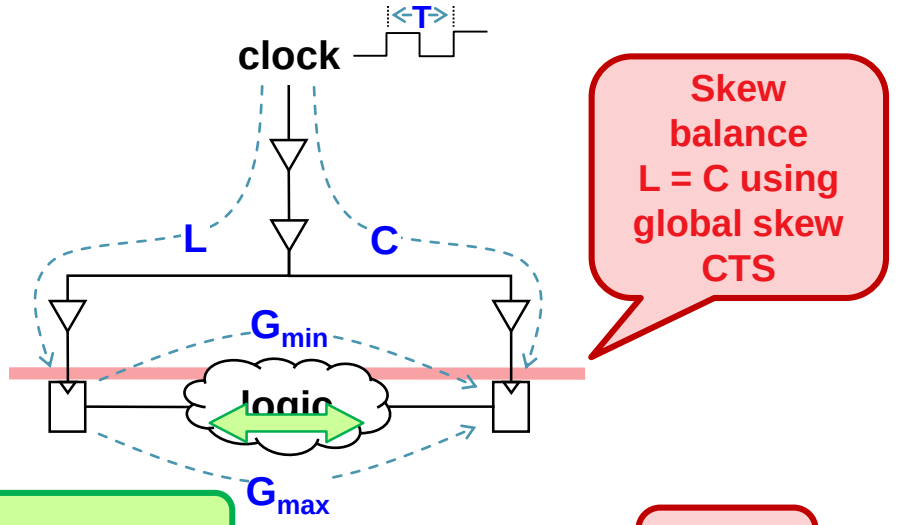






cadence

Clock Design Flow (Traditional vs CCopt)



Skew balance L = C using global skew CTS

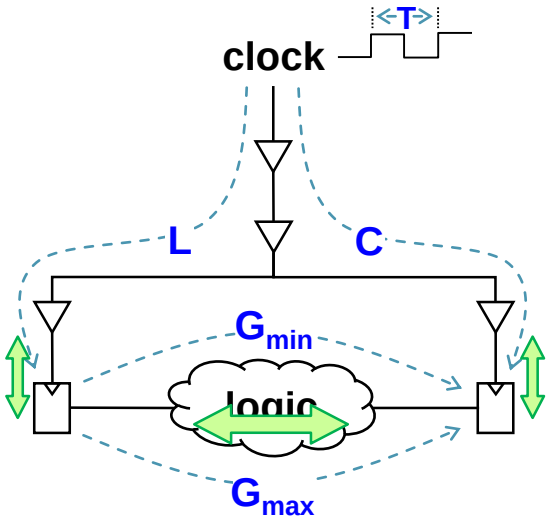
VARIABLE

FIXED = 0

Setup: G_{max} < T – (L – C)

Hold: G_{min} > (C – L)

local skew



Schedule clock delays based on timing slacks

VARIABLE

VARIABLE

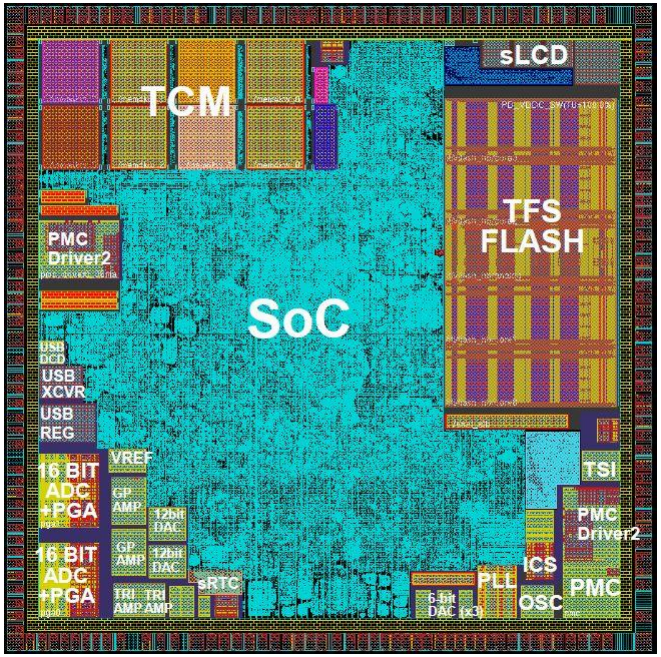
Setup: G_{max} < T – (L – C)

Hold: G_{min} > (C – L)

local skew

cadence

Customer Case: Low-Power 32-bit MCU Implementation



Low power techniques used:

- Clock gating
- Multi-vi
- Extended gate-length
- Back/Source biasing
- PSO
- Multi-power-modes
- MSV
- Multi-bit SRPG
- On-chip voltage regulator
- Shifter @ Vdd & Vss
- Fine-grain Voltage Scale
- Adaptive biasing
- Low swing clock

Dynamic power reduced by 30% and static power by 80% in the off state.

Advanced Low Power Techniques

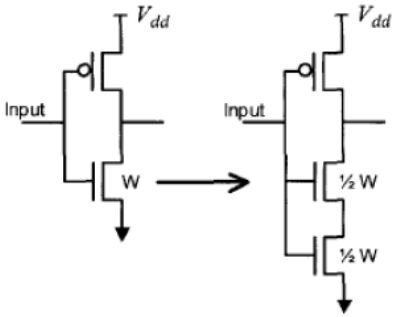
Approaches for Greener IC

$$P = C\ V^2\ f + V\ I_{(static+overlap)}$$

- EDA/Circuit Design Techniques
- Reduce Leakage (energy efficient)
  - Ideally, if leakage ~= 0, no need to do PSO
    - No need for isolation (less area, less trouble)
- Reduce Vdd Voltage (lower dynamic power)
  - **V^2** power reduction (>= 10x)
  - Most appealing way! (as area, frequency going way up)

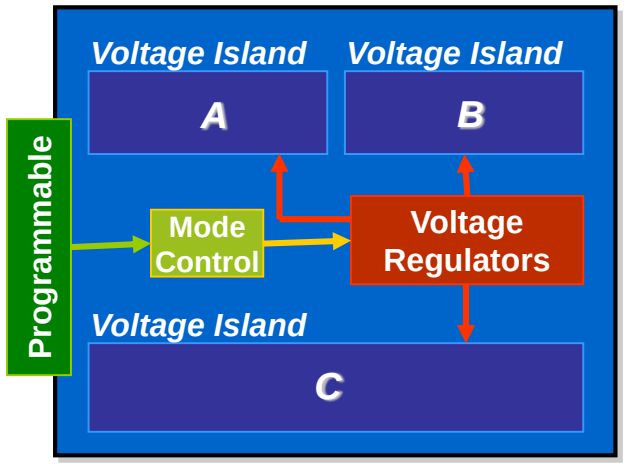
Review: Traditional Leakage Power Optimization

- Multi-Vt Swapping
  - HVT -> lower leakage
- Multi-Gate-Length Swapping
  - Longer gate-length -> lower leakage
- Stack-Forcing
  - Increase effective gate length for leakage reductior
- Poly-biasing
  - Increase gate-length at GDS level



Dynamic Voltage Frequency Scaling (DVFS)

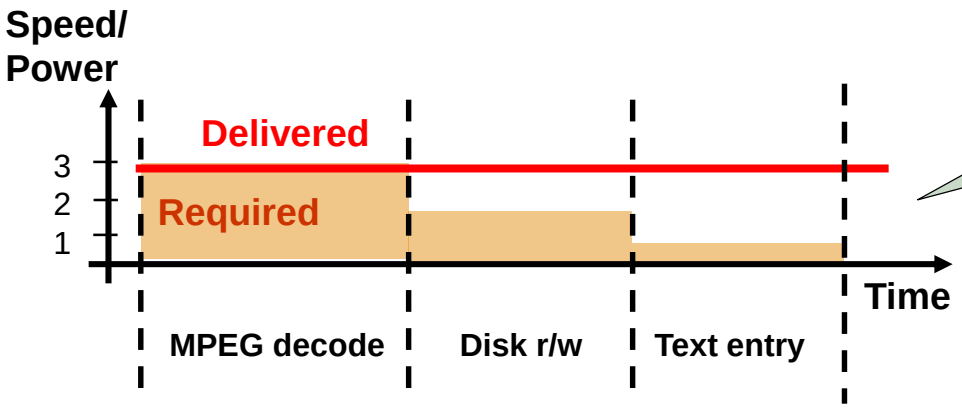
- **Description**
  - Varies the frequency and voltage of a design
  - Done Real time
  - Commonly used in processor design
  - **Based on system demand**



- **Power Savings**
  - Optimal voltage/Frequency level per task per domain
  - Improves both dynamic and leakage voltage
    - Reduced frequency – produces less switching power
    - Reduced voltage means
      - Less dynamic power
      - Less leakage power

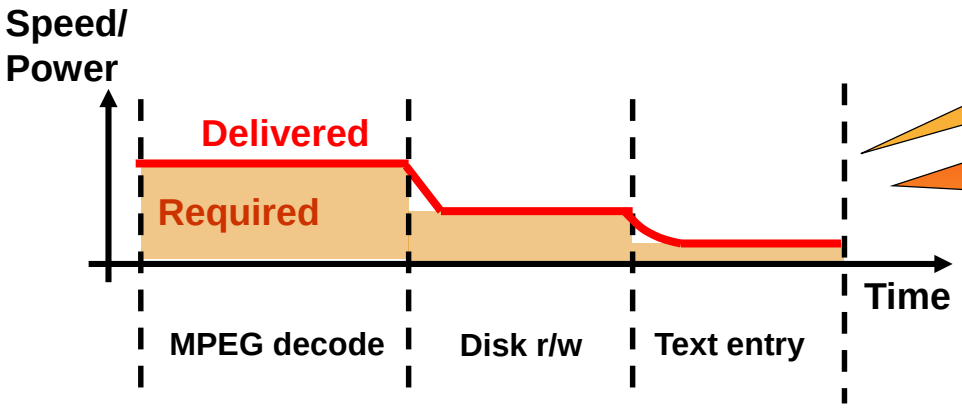
Mode	Domain A	Domain B
High Perf.	1.2V 800mhz	1.2V 600mhz
Med Perf	1.0V 600mhz	1.2V 600mhz
Idle	0.8v 400 mhz	0.8v 400mhz

Dynamic Voltage Frequency Scaling (DVFS)



Without DVFS - deliver same power in all modes 3W average

DVFS provides a better match to requirements and can save power over time



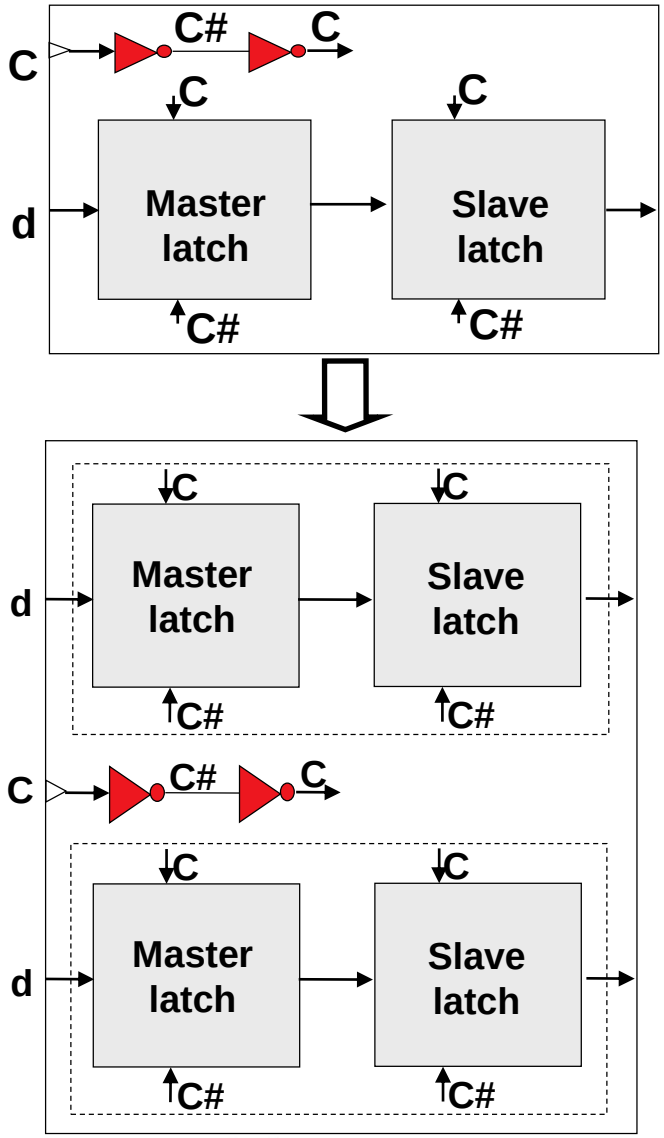
2W average power in this diagram, but varies based on system requirements

cadence

Dual/Multi-Bit Flops

- aka: Multi-Bit-Cell-Inference (MBCI)

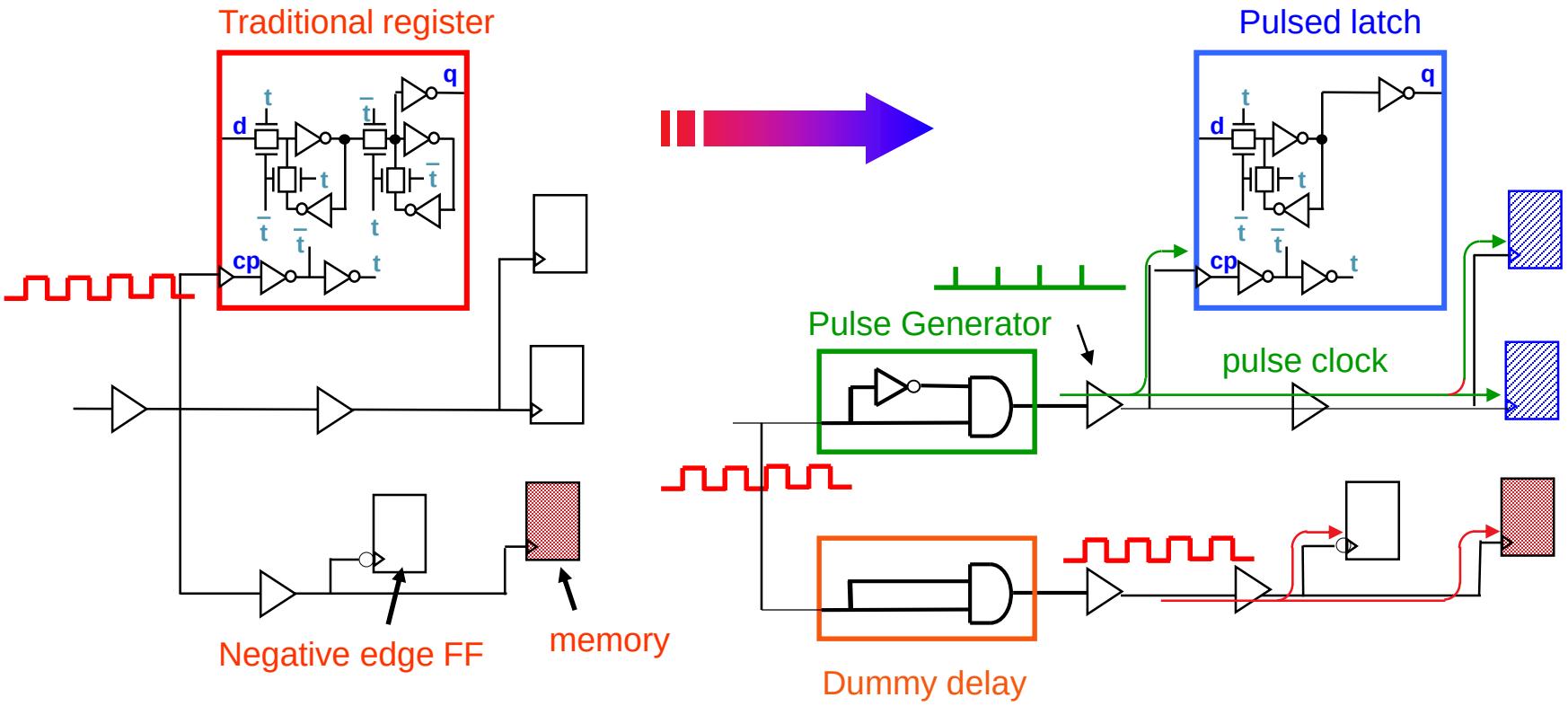
- Inverters in flops tend to be oversized due to manufacturing ground rules.
- As we get into smaller geometries like 65nm and beyond the minimum size of clock driver can drive more than single flop.
- Combining registers into multi-bit instances reduces the total load on the clock tree.
  - Dual-bit or Quad-bit flops are designed to efficiently distribute the internal clock signal to the master & slave elements of the flop.
- By using this we can reduce the leaf load on clock tree by a max 50%.



cadence

Pulsed Latch Design Methodology

- Traditional FF is replaced with a pulsed-latch
- Pulse generator is shared by several pulsed-latch
- Dummy clock delay cell is used to balance clock tree



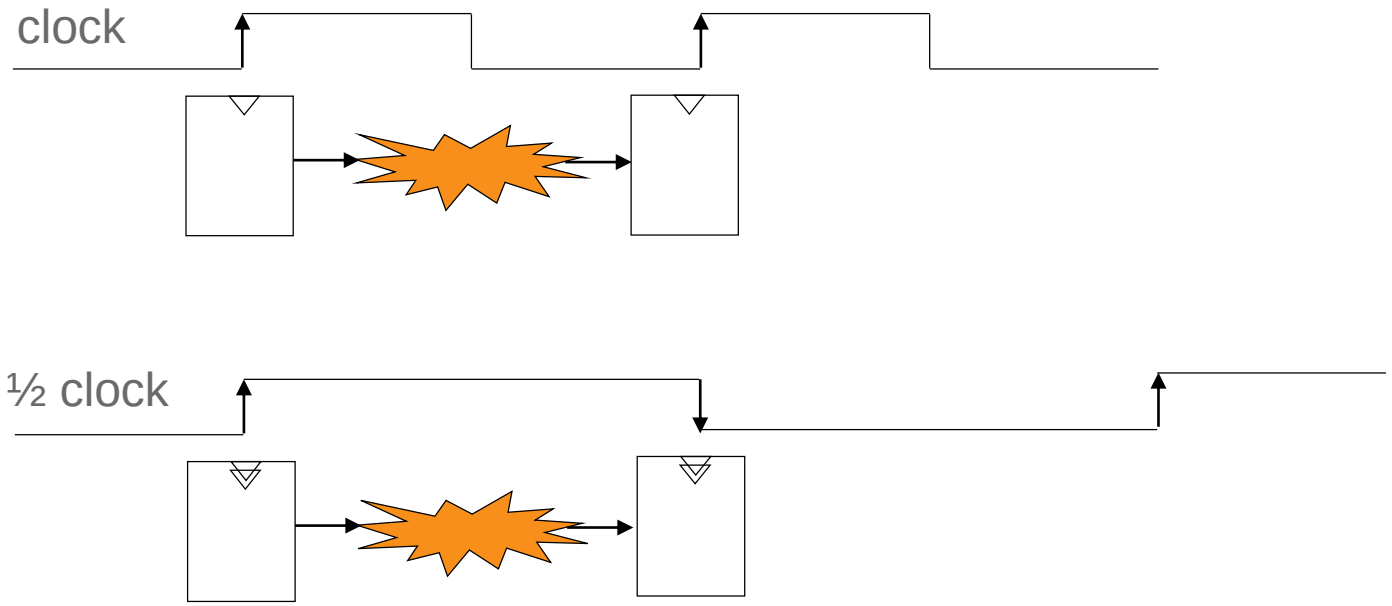
cadence

Other Techniques to Reduce Clock Power

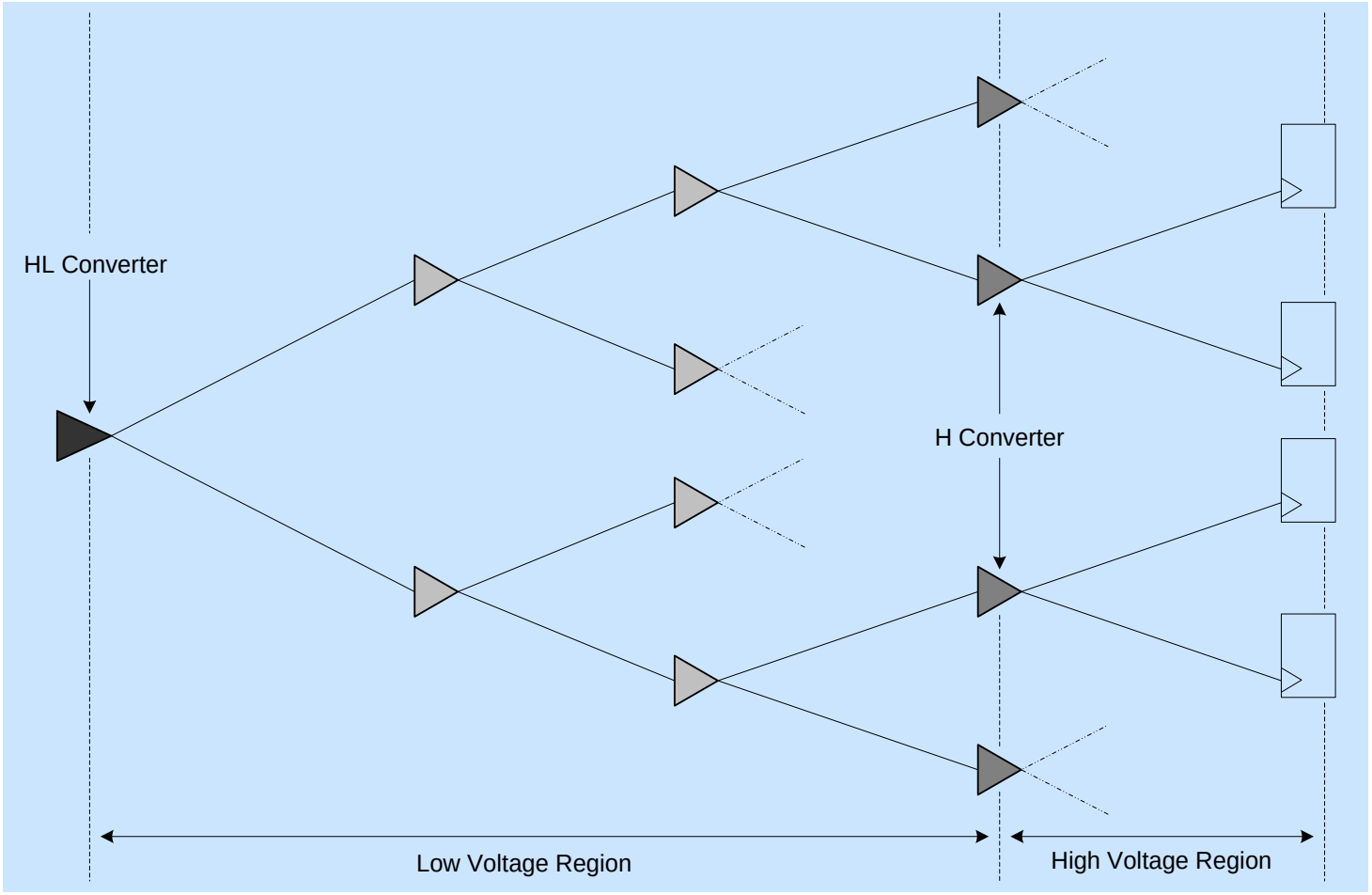
- Dual-Edge-Triggered FF (DET)
  - Double data rate per cycle -> half the clock frequency
- Low Swing Clock (LSC)
  - Use low voltages for clock tree
- Clustered Voltage Scaling (CVS)
  - Extend LSC idea to data path w/ slacks
- Globally asynchronous locally synchronous (GALS)
  - No more global clock

Dual-Edge-Triggered Flip-Flops (DETFF)

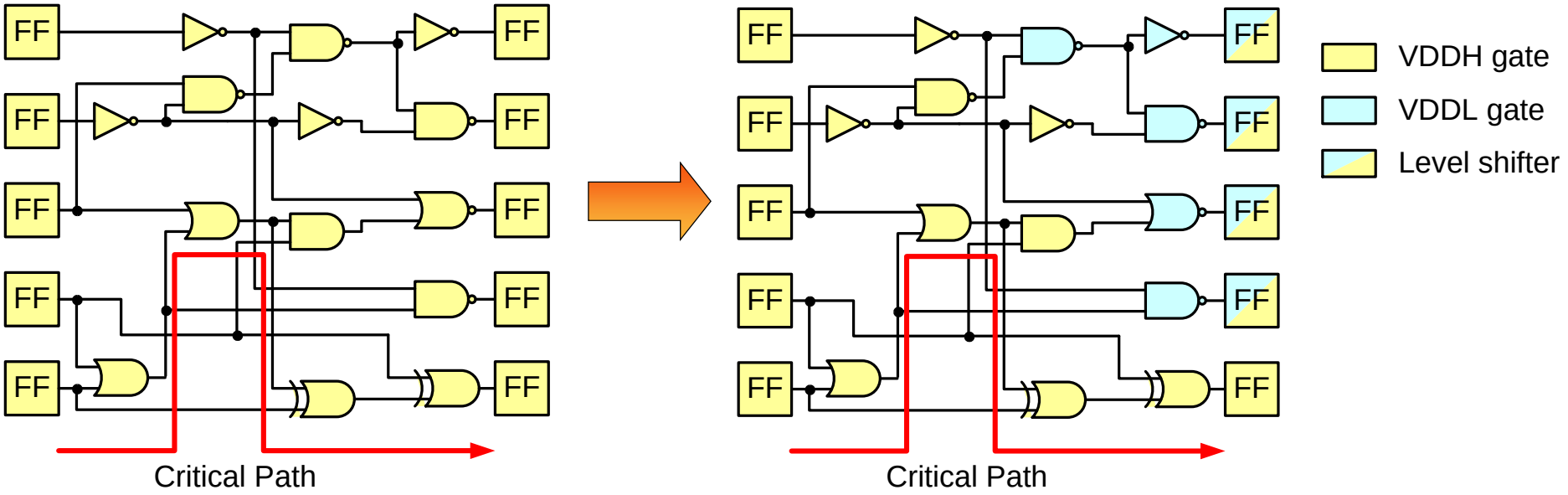
- Require special FF design (IP)
- 50% duty cycle not easy to achieve
- Challenges in skew reduction or useful skew optimization
- Timing verification / Multi-cycle path handling



Low Swing Clock Design

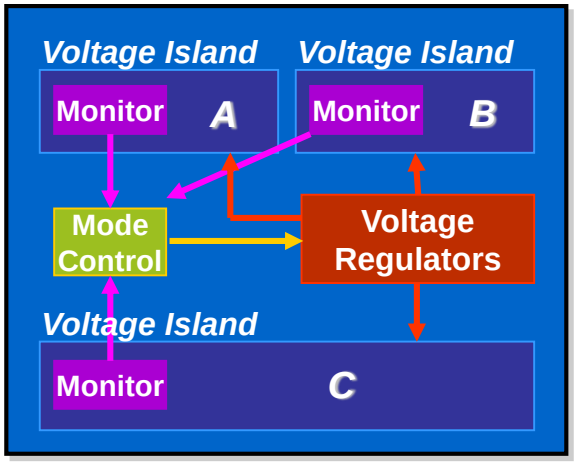


Clustered Voltage Scaling (CVS)

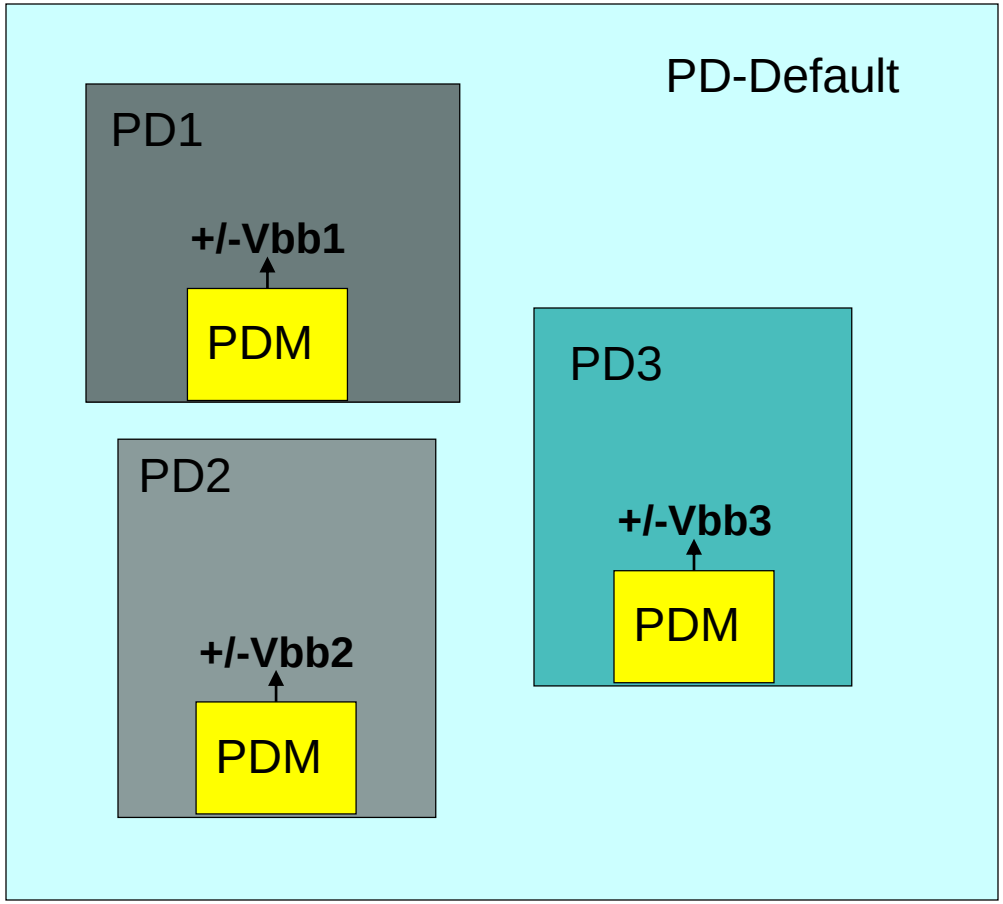


Adaptive Voltage Scaling (AVS)

- **Description**
  - A finer level of control version of DVFS
  - Utilizes Hardware monitors as input
    - Monitor PVT, load, timing
    - Adjusts frequency to compensate for on chip variations, temperature, etc
  - SW control for major modes, but HPM to fine tune
  - Typically used only in very high end designs like processors.
- **Power Savings**
  - Provides the best potential savings as every domain is dynamically tuned to its environment
  - Voltage increments can be as small as 0.01V increments
- **Advanced Technique**
  - This technique is the hardest to implement and verify
    - Largest overhead of circuitry
    - Can lead to huge verification challenges



Adaptive Back Bias Vt Control



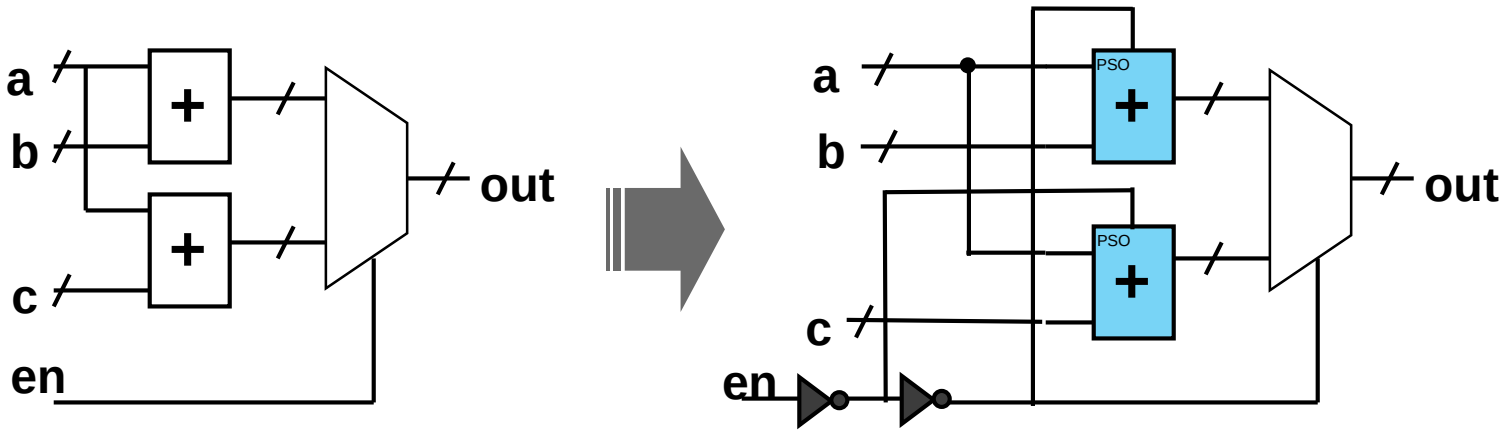
- **Concept applicable to AVS as well**

Leakage Power Optimization:

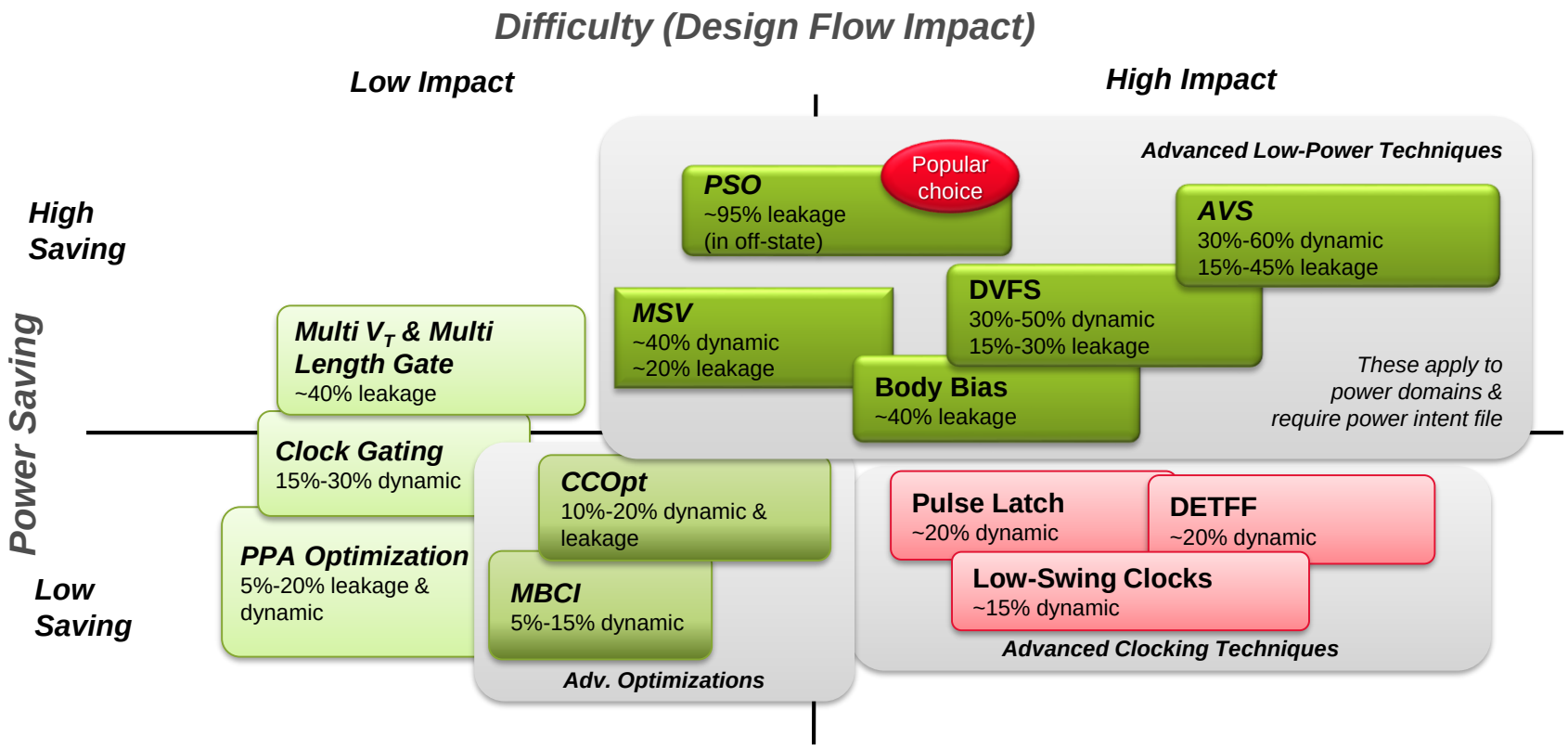
- Turn off power to unused logic trees
- Operand isolation recognizes logic cones that are blocked
  - Extending this idea to Power Shut Off, these unused logic cones could be powered off

```
module test (en,a,b,c,out);
input en;
input [7:0] a, b, c;
output [8:0] out;

assign out = en? a+b : a+c;
endmodule
```



Relative ROI – Low Power Design Techniques

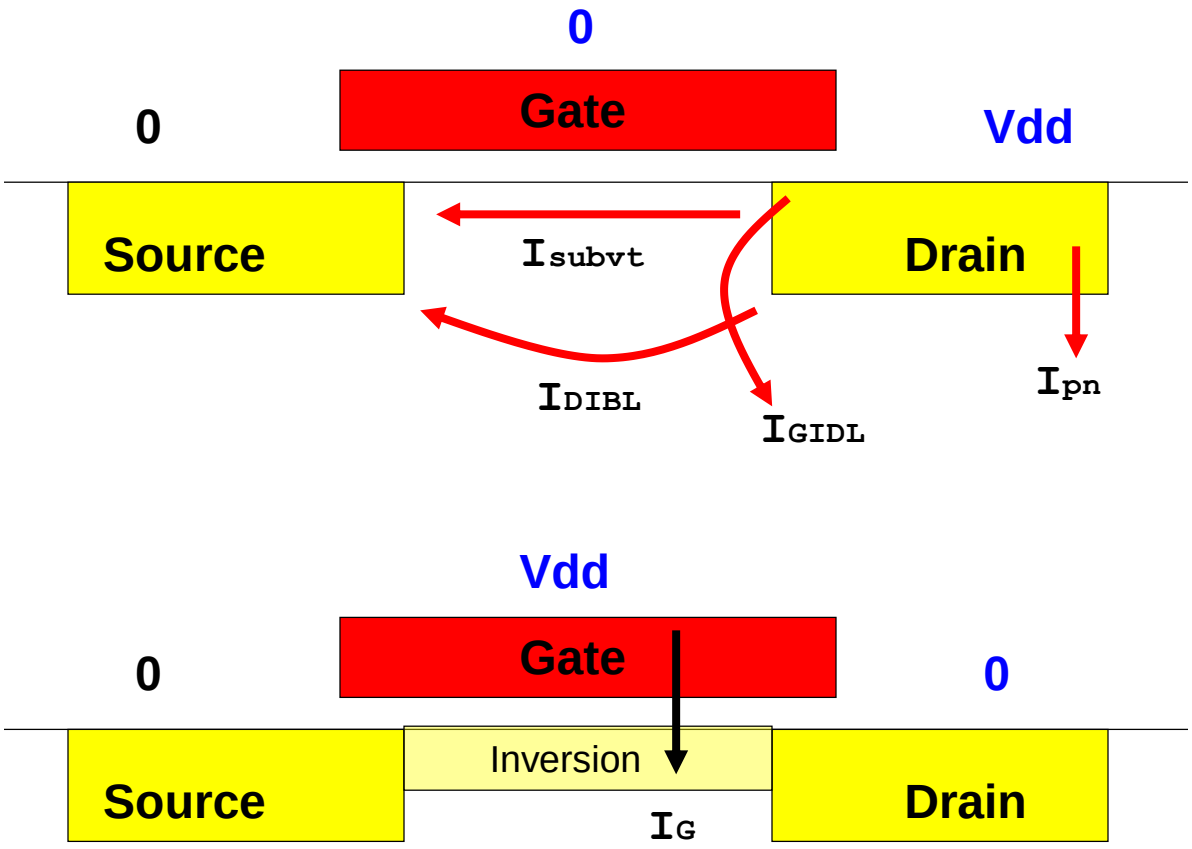


Devices – Now & Future

Strained-Si, Hi-K/Metal Gate

- Reduce gate leakage (I_G)
  - Thicker gate oxide (1.2nm->3.0nm)
    - Harder for electron to tunnel directly through gate
  - Need to compensate for speed lost
    - Strained Si to improve mobility at channel surface
    - Hi-K Dielectric Material (> 3.9)(15?) + Metal Gate
      - Actual K value is a trade secret
      - 5-100x reduction in I_G

Review – Small Geometry Leakage Components

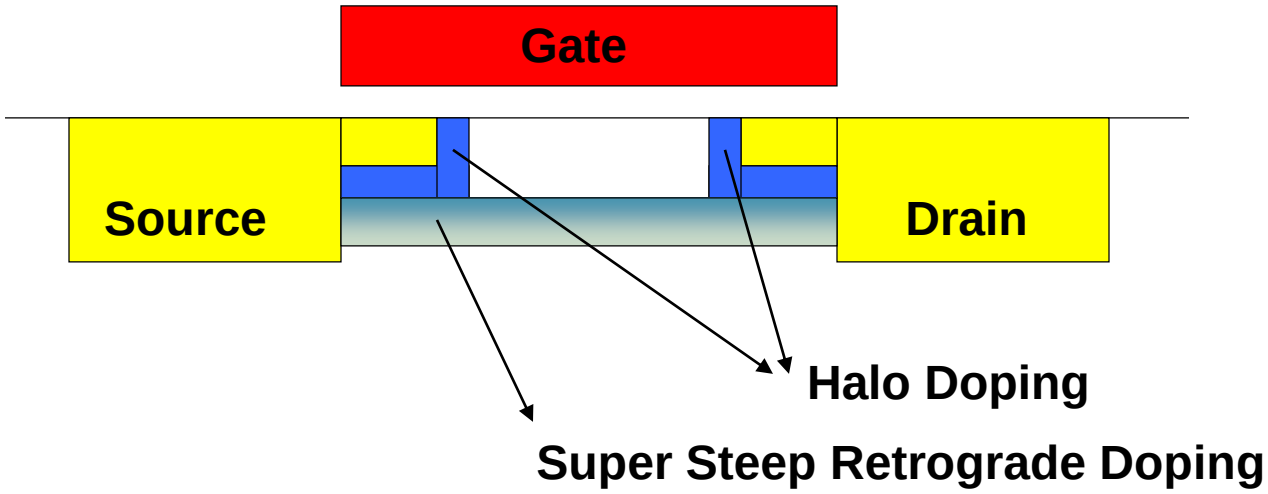


* I_G : direct tunneling + hot carrier injection

DIBL: drain-induced barrier lowering
GIDL: gate-induced drain leakage

Well / Channel Engineering

- Reduce channel (surface & under) leakage due to DIBL
- Improve Vt sensitivity to body bias
  - Allow adaptive Vt modulation by body bias (I_{subvt})



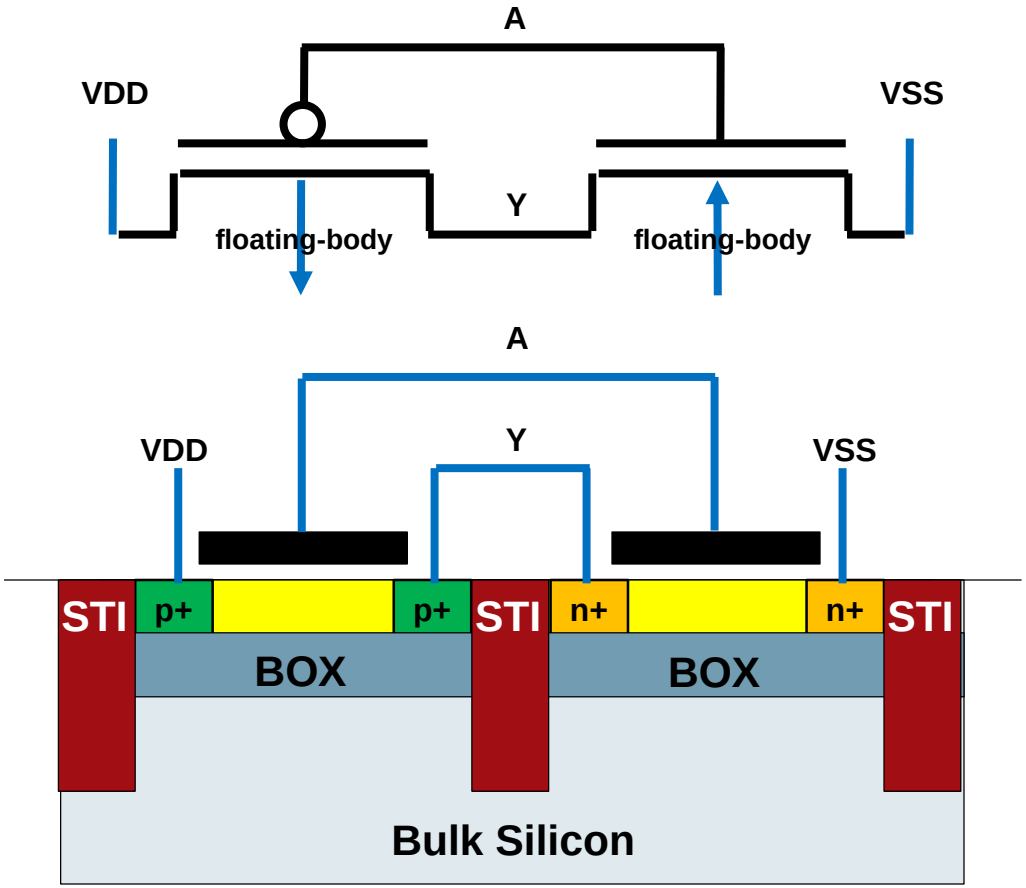
SOI & FinFET

cadence™

Why SOI

- Traditional bulk process limitations (28nm/14nm)
  - DVFS
    - Voltage limited & performance degradation
  - Poly biasing
    - Limited range
  - Dynamic transistor Vt control
    - Limited body bias range (-300mV to +300mV)
    - Limited benefit in 28nm, no benefit beyond 28nm
- SOI
  - No latch up
  - No parasitic device
  - Low soft-error due to alpha-particles
  - No channel doping -> improve Vt variability (Vt can be much lower-> Vdd too!)
  - Ultra-thin insulator -> large back-bias voltage possible (no GIDL)
  - Ultra-thin body & buried-oxide (UTBB) -> better short-channel-effect

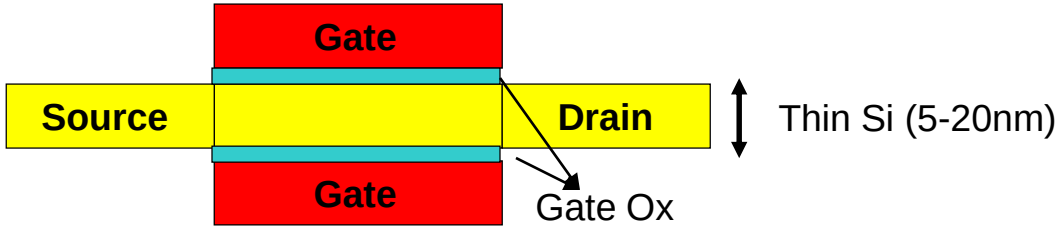
SOI Fully-Depleted Process (fundamental)



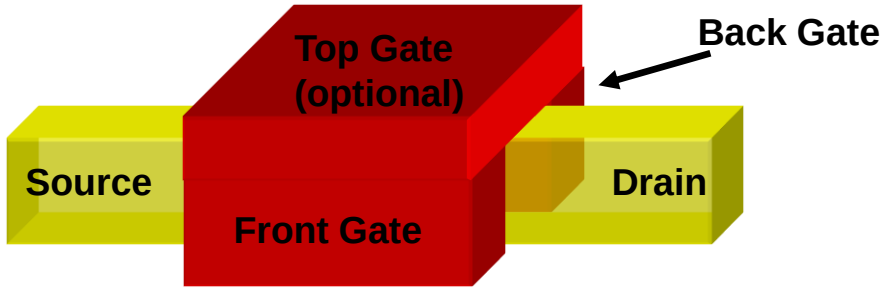
* BOX: buried oxide
* STI: shallow trench isolator

FinFET / Multi-gate FET

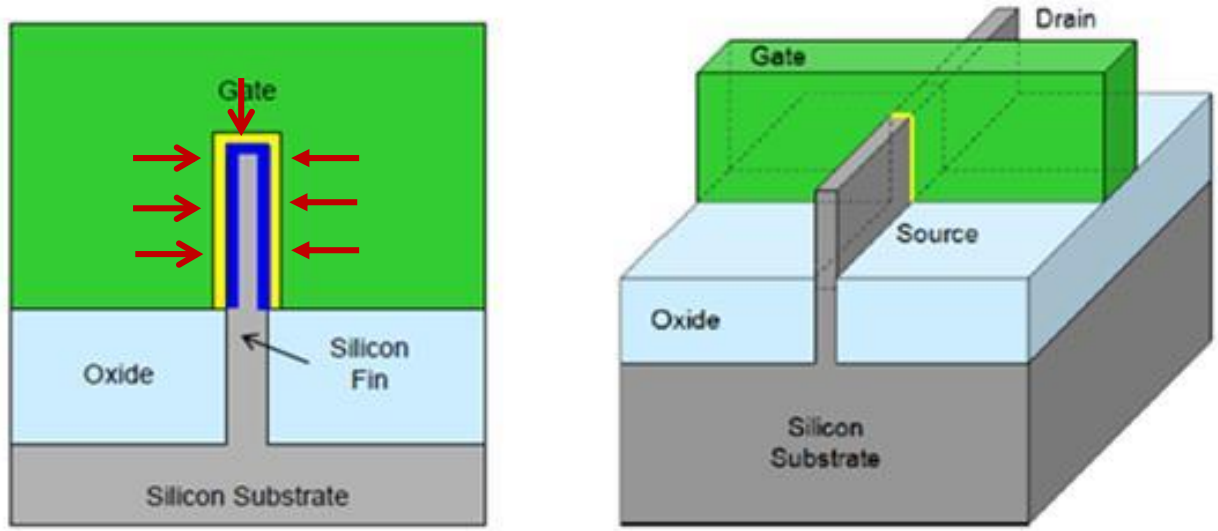
Top View



3-D View



FinFET



- Gate control silicon fin from three sides:
  - Smaller leakage power (15%~25%)
  - Better driving strength. Can use lower Vdd to achieve the same performance => Better Dynamic Power.
- **Variability will be more serious due to difficult control on Fin h/w.**

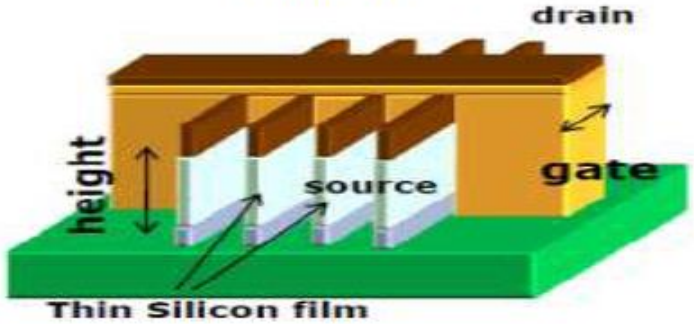
SOI vs FinFET – Comparison Scores (Biased)

Comparison	FDSOI	FinFET
Metal stack, design methodology		
Leakage mitigation	1	
Dynamic power mitigation		2
Supply chain		1
Density		1
Manufacturability	1	
Variability	1	
Performance		2
Design Portability	1	
SRAM	2	
Analog		
Future scaling		1
Total	6	7

Source: Designing SoCs with Planar Fully Depleted Device and FinFETs, Rob Aitken, ARM (Fully Depleted Transistors Technology Symposium, Dec. 10, 2012, San Francisco)

SOI vs FinFET – Comparison 2 (Biased)

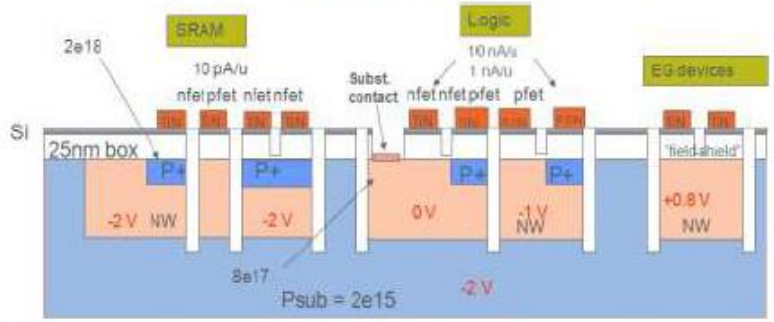
FINFET



- Best electrostatics (DIBL, SS)
- Highest Drive Current (per unit area)
- Higher Ceff (including high Miller component from region between FINS)
- Sources of variability (Dfin, Hfin, Fin taper)
- Undoped/low fin doping → good RDF
- Quantized active width – but better active efficiency in standard cells (improved PPA)
- Higher process complexity

Most suited for high performance applications

FDSOI



- Good electrostatics (DIBL, SS)
- Lower drive current due to simpler process
- Lower Ceff
- Sources of variability (Tsi, Tbox)
- Undoped channel device -> better RDF and device matching
- Lower process complexity offsets higher substrate cost
- Back Bias management is critical (back gate doping, power routing)

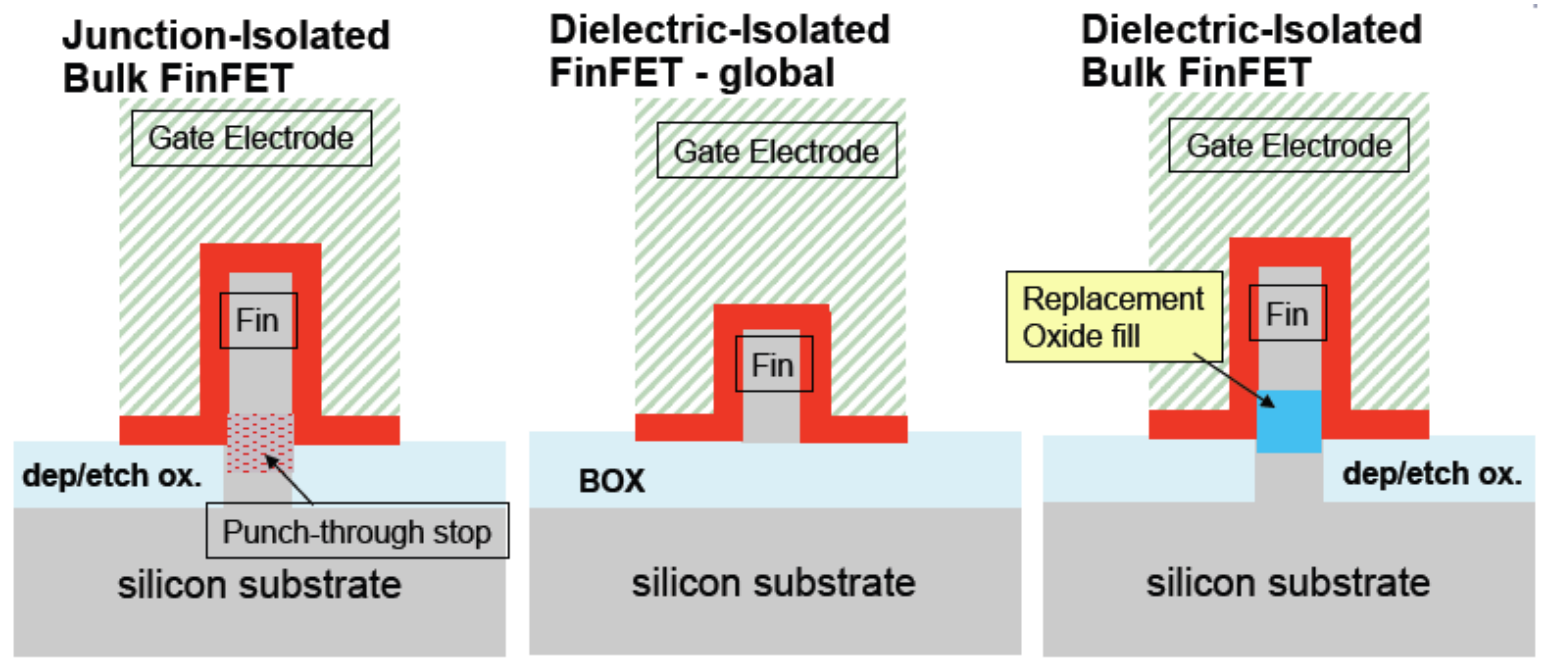
Most suited for low power / low leakage applications

2

Source: SoC Differentiation using FDSOI – a Manufacturing Partner's Perspective, Shigeru Shimauchi, GlobalFoundry (FD-SOI Workshop, June 15, 2013, Kyoto)

cadence

Future – FinFET on Oxide (FOx) ???



Fin on bulk

- Fin height controlled by ox etch
- Complex isolation scheme
- Fin doping control possible issue
- Uses bulk wafer

Fin on SOI

- Fin height set by substrate
- Simpler isolation scheme
- Requires SOI wafer

Fin on oxide over bulk

- Doping isolation requirement removed
- Isolation scheme still complex
- Uses bulk wafer

Source: 2nd Generation FinFETS and Fins on Oxide, Ed Nowak, IBM
(Fully Depleted Transistors Technology Symposium, Dec. 10, 2012, San Francisco)

cadence

Advanced Devices

cadence™

ITRS 2.0 Logic Voltage Roadmap (2015)

ITRS-2.0	2015	2017	2019	2021	2024	2027	2030
Node(GL)	16/14(28)	11/10(22)	8/7(18)	6/5(14)	4/3(11)	3/2.5(9)	2/1.5(7)
VDD	0.80	0.75	0.70	0.65	0.55	0.45	0.40
Device	FinFET FDSOI	FinFet FDSOI	FinFET LGAA	FinFET LGAA VGAA	VGAA M3D	VGAA M3D	VGAA M3D

LGAA: lateral gate-all-around; VGAA: vertical gate-all-around; M3D: monolithic 3D IC

- Voltage does NOT scaled linearly w.r.t. node(gate-length)
- Bottleneck: variation & reliability
  - Random dopant fluctuation (RDF)
  - Static noise margin (SNM)
- Minimum voltage requirement (Vmin)

Source: International Technology Roadmap for Semiconductors 2.0 Executive Report 2015, pp 34

For 10nm and Below

- Equivalent Scaling (traditional – maintain constant E-field)
  - Strained Si (90nm)
  - High-K Metal Gate (HK/MG – 45nm)
  - FinFET (22nm)
  - Non-Si: Germaniun
- 3D Power Scaling (future <=10nm)
  - Monolithic 3D (running out of horizontal space)
  - Combination of 3-D architecture (FinFET) & low power device
  - Gate-all-around (GAA): Lateral (LGAA), Vertical (VGAA)

Source: International Technology Roadmap for Semiconductors 2.0 Executive Report 2015, pps 2,32-38

Device Natural Length

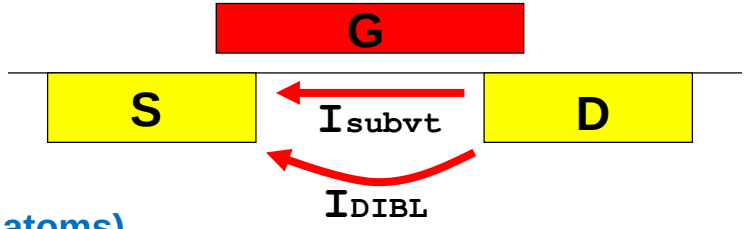
- Natural length
  - The lateral distance of E field that drain can influence (Vt) under the gate channel area
  - The longer the worse!
  - Traditionally (bulk Si) gate length was chosen to be 4x to 6x of natural length

$$\lambda_n = \sqrt{\frac{1}{n}\frac{\varepsilon_s}{\varepsilon_{ox}} W t_{ox}}$$

Gate-oxide ~= 1.2nm (~= 5 Si atoms)

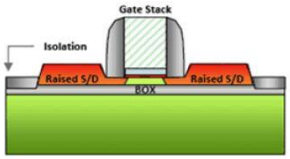
Hi-K

n = number of gates of device

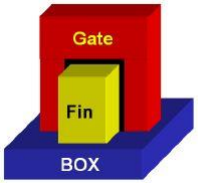


- Gate-all-around (GAA) to increase n to reduce natural length

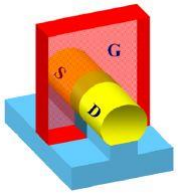
Natural Lengths for Various Devices



2D Bulk/SOI
n=1



3D FinFET
n=2..3



LGAA (horizontal)
n=3

VGAA (vertical)
n>=4

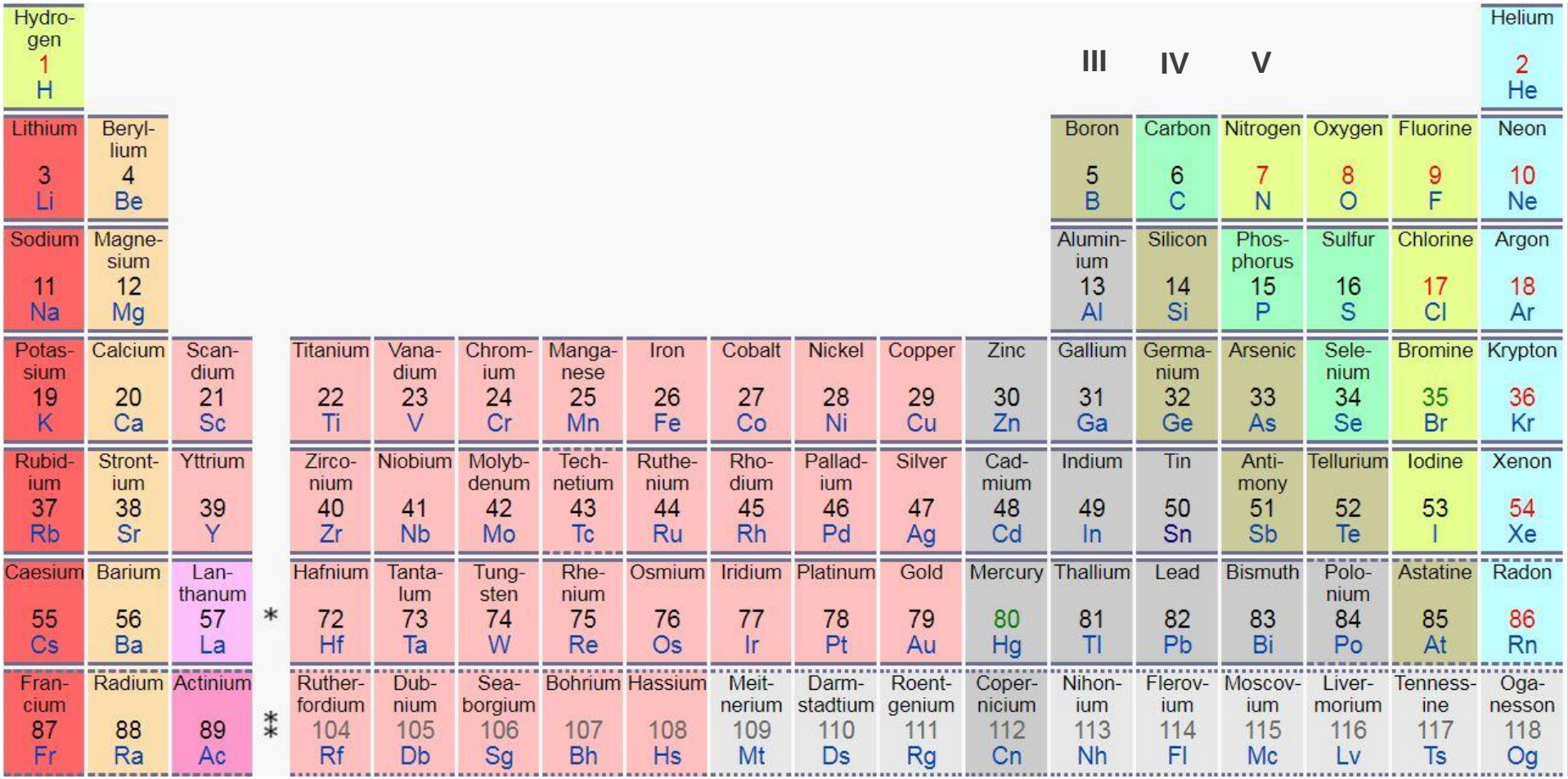
- Practical gate-length limit
  - FinFET: 8nm
  - GAA: 3nm has been demonstrated (Ansari et al, Applied Physics Letter, 2010)

New Materials, New Physics & Quantum Effect

- Ultra-low-power applications
  - Operating voltage below Vt
    - Ie, leakage current (today) = operating current (future)
    - subthreshold slope 60mV/dec thermal voltage limitation
  - New (non-Si) Materials & Non-MOS-like devices
    - Eg, no PN-junction
  - Quantum effect governs at extreme small dimension (smaller than FinFET)
    - Eg. Tunneling
- Examples and active research topics:
  - **Nanowire transistor (NWT)**
  - **Carbon-nanotube transistor (CNT)**
  - Junction-less transistor (JNT)
  - III-V compounds: GaSb-InAs, GaAs, …
  - Combination of group IV: **Graphene**, Ge-Sn, …
  - Tunneling FET (TFET)
  - Metal-Semimetal-Metal (all same material w/o PN junction, eg Sn)
  - Band-gap engineering (extreme small dimemsion widens the metal band-gap!)

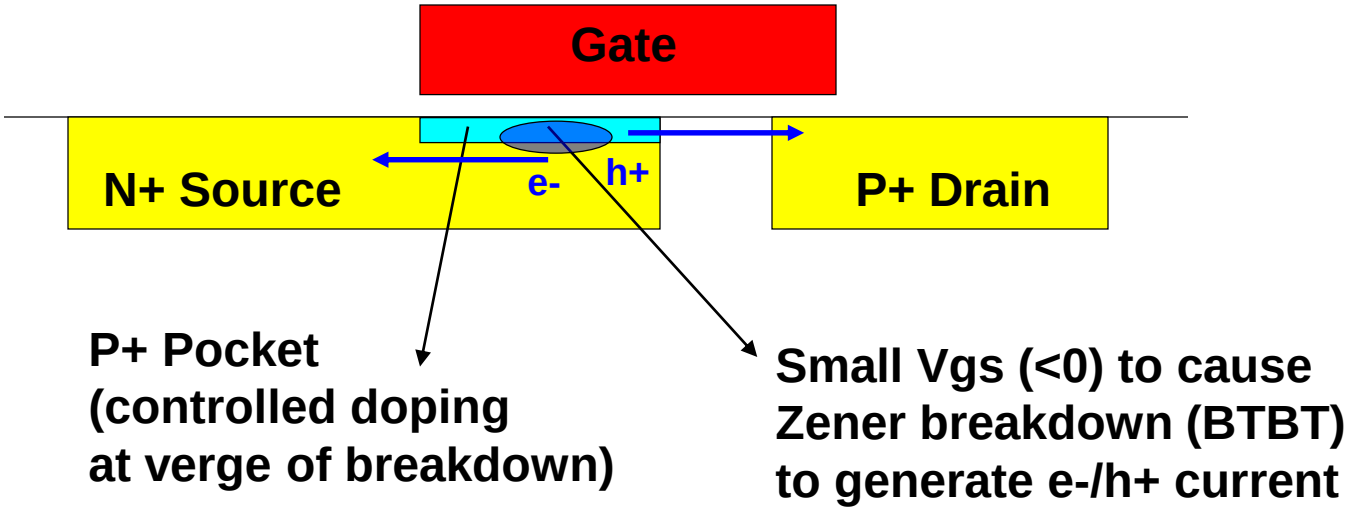
(All of the above are formulated w/ Poisson and Schroedinger eq. and can be solved w/ perturbation approach)

Periodic Table (Partial)



Research Example: Low Voltage (Green)FET

- Based on Band-To-Band Tunneling (BTBT)
  - Identical to GIDL mechanism (leakage -> useful current)
  - Rely on carrier going through (instead of over) barrier
  - Operate at Vdd ~= 0.2V (10x power reduction)
  - Experimental, no Si yet



* "Green Transistor - A VDD Scaling Path for Future Low Power ICs," C. Hu et al, VLSI-TSA 2008

Research Institutes for Ultra-Low-Power Devices

- STARnet
  - Semiconductors Technology Advanced Research network
- NRI
  - Nanotechnology Research Initiative
- LEAST
  - Center for Low Energy Systems Technology

What Next?

Traits of Digital IC Design Automation

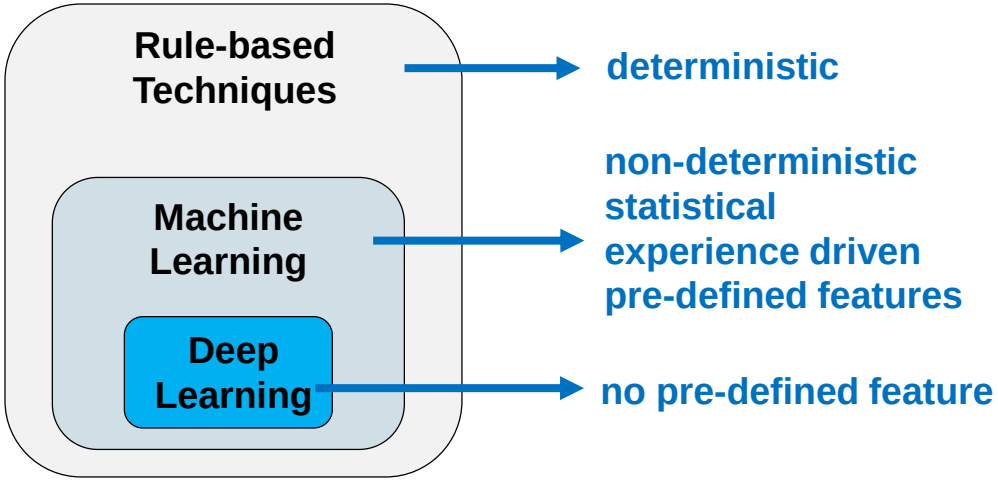
- **What's involved:**
  - Large problem size
    - b/millions of gates
  - Statistical in nature, high dimensionality, high-order interactions, discontinuities, non-linearity, variations & uncertainty
  - Optimization
    - Performance, power, area
  - NP-completeness
  - Heuristics, Estimations, & Predictions
    - Congestion, power, delay, extraction
  - Heavily depends on design knowledge & experience

- **What comes to the rescue?**

Source: Semiconductor Engineering

Artificial Intelligence (AI) in EDA

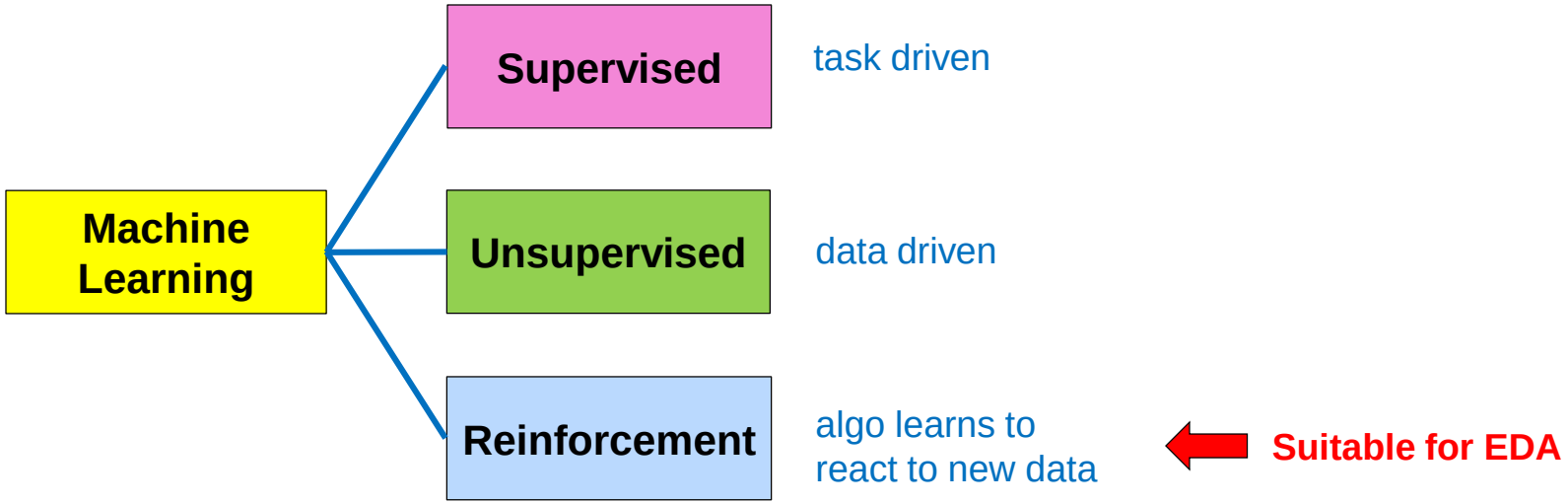
- **AI is no longer Sci-Fi, it is around us now!**
  - Machine Learning (ML)
  - Deep Learning (DL)
- **Leverage design experiences via training**
  - Generate designs
  - Input data reduction
  - Optimal design decisions & flow
  - Making new observations & derive/infer new features/rules



Source: Semiconductor Engineering

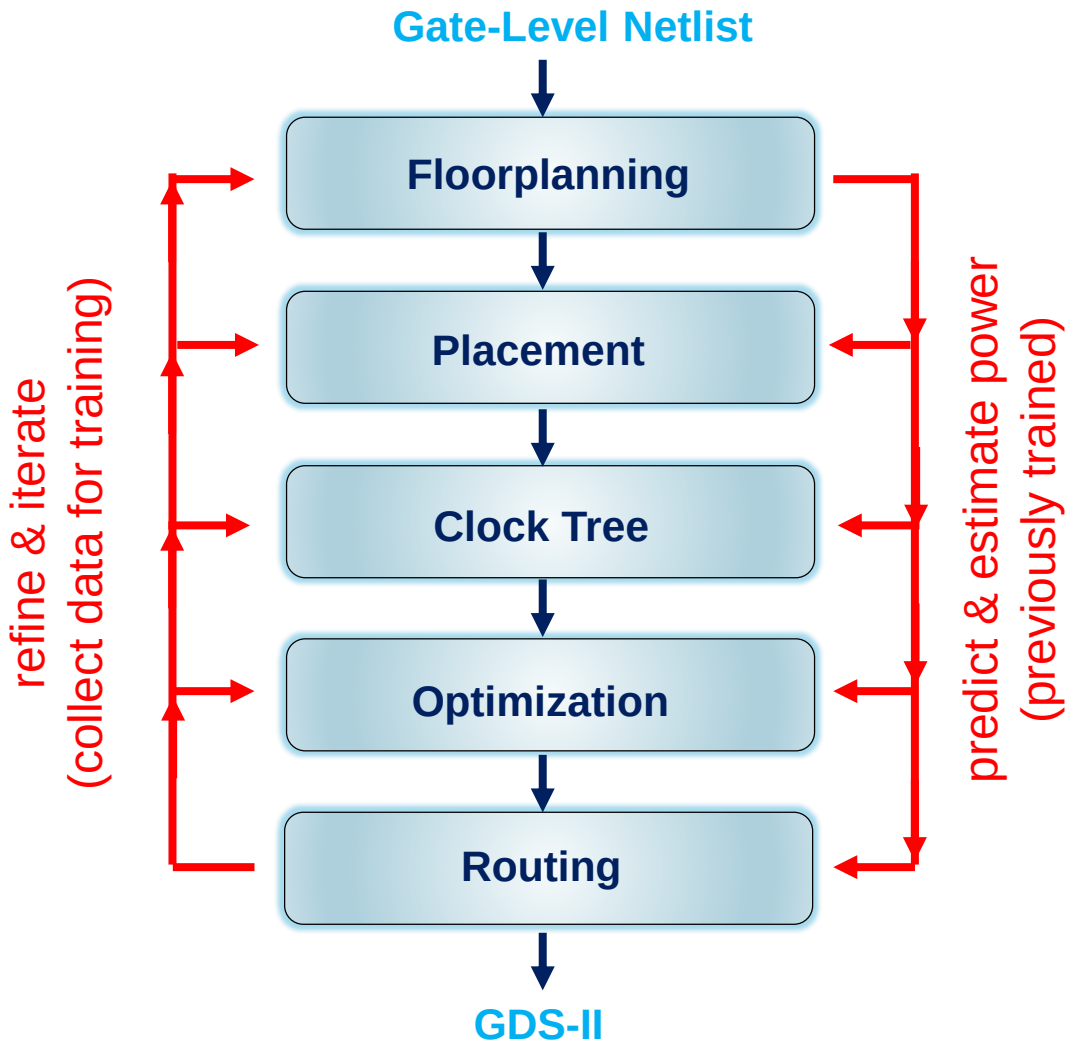
Machine Learning

- **Use past experience to solve similar problems**
  - With training dataset, learn to predict on new data for better design optimization



Source: Semiconductor Engineering

Design Flow & Power Optimization w/ ML



Power related features & metrics:

- Domain shapes
- Macro/Mem locations
- Interconnect lengths & caps
- Buffer usage
- CV^2f on nets
- …

cadence

Challenges of ML in EDA

- **Lack of training dataset**
  - Data not representative, results not accurate
  - Designs / type of designs are different
  - Technology dependency: old nodes vs new nodes (different constraints)
  - Type of designs are different
  - Customers won't give you data for training!
- **Assessment of risks**
  - Non-deterministic output, hard to debug
- **Cost of training & How to commercialize**
  - How many GPUs, training data collection/generation, training time, …
  - Business models (how to make $)

Source: Semiconductor Engineering

Summary

Low power is the future, and the future is now!

$$P = C\ V^2\ f + V\ I_{(static+overlap)}$$

- Minimize P ! (eg. if V is fixed, minimize C*f, not C or f alone)

- Think high level - the higher the level, the better the ROI
- Power(energy) must be part of cost consideration at all levels of design
- Beware of what future technologies can bring, and be prepared for them
  - Due to technology advancement, some techniques may not have good ROI
    - Eg. with FinFET, leakage optimization might not be necessary
  - Scaling: beware of Reliability, Variability, & Static Noise Margin (SNM)
- AI/ML/DL techniques to help optimizing power


cadence®

Appendix

Low Power Techniques at Other Levels (the Global Picture)

Register-Transfer Level

- State assignment / encoding, use of DC
- Multi-level power-aware logic transformation/optimization
- Pre-computation (Shannon's expansion)
- Operand isolation / data gating
- Global bus splitting & partitioning / Dedicated bus
  - Bus-splitter / router (NoC)
- Power-off unused units
- Factoring
- Operation substitution / reduction (while maintaining throughput)
- Technology mapping (choice of cell)
- Choice of components
  - Eg. Ripple-carry instead of CLA: slower but lower power
- Pipelining (scale the voltage on the pipelined blocks)
  - Same throughput, less power
- Parallel Processing (scale the voltage on the parallel blocks)
- Memory reorganization (split up memory lookup)
- Datapath Reordering (glitch avoidance)

Architectural Level

- Asynchronous design, GALS
- Multiple power modes (degree of “darkness”)
  - Active, alive, drowsy, nap, doze, sleep, (light/deep/deeper) sleep, off, dark, dim, …
- On-Offfff-On-Offfff-On operation instead of AO (requires profiling)
- Approximate Computation (imperfect, inaccurate,…)
  - Qos select controlled by HW/SW/user
- Redundant/Parallel computation units – HP & LP version
- Domain-specific accelerators
- Minimize memory access
- Minimize number of operations (*, +)
- High-level resource allocation & scheduling
  - Minimize data movement
- Proper representation of data
  - Use sign-mag instead of 2’s comp if appropriate (eg, when sign changes frequently)
  - Gray coding (if data bits changes sequentially – eg. instruction memory address)

System/App/Software Level (some examples)

- Rework software to minimize power
  - Choice of Algorithm
    - If can do in O(n log n), don't do in O(n^2)
      - Eg. Proper vector quantization (VQ) algorithm – differential tree search instead of full
  - Remove redundant activities in code
    - Eg. refresh rate 60Hz of the same screen
    - Eg. frame update = 60 FPS of the same frame
    - 30%-40% power saving (DAC 2014), similar image quality
  - Generate low power version of the original image
    - Partial display disable/dim
    - Color remap
    - Substantial power saving (configuration dependent) (DAC 2014)
- Activity monitoring & activity-based SW DVFS control
  - Eg. either CPU is bottleneck or GPU is, but not both
    - DVFS on CPU or GPU depending on the current activity
    - 30% power saving on average (DAC 2014)
- Dark Silicon
  - Dynamically configure/program the NoC "ON-OFF" routers & blocks by software
    - Substantial power saving (DAC 2014) (algorithm dependent)

Circuit / Cell / Memory / IP Level

- Multi-Stacking / Multi-Gate-Length / Multi-Vt
- Multi-supply / Multi-rail
- Back-bias support (FBB / RBB)
- Minimize internal node glitches
- Minimize internal node cap
- Fast internal node slew rate
- Use static circuit & minimize dynamic circuit & pre-charging
  - Eg. Schmitt-trigger inverter based gates (ISSCC 2011)
- Combine logic & latch in single cell (eg. Alpha latch)
- Memory / IP
  - Built-in multiple power modes with power management unit
  - Smaller banks, decoder logic
  - Shorter bit-line, Word-line under-drive (unselected cells remain at low voltage)
  - Internal DVFS
  - Result caching (prevent redundant lookup – the core can stay at low voltage)
- Analog Components
  - PLL/DLL partitioning (phase detector sharing among partitions)
- Post-Si Calibration/trimming, Poly-biasing