

# Digital IC Design

## Lecture 6:

### Low Power Design Techniques for DIC

黃柏蒼 Po-Tsang (Bug) Huang

bughuang@nycu.edu.tw

International College of Semiconductor Technology  
National Chiao Tung Yang Ming University

國立陽明交通大學

NATIONAL YANG MING CHIAO TUNG UNIVERSITY

# CMOS Energy & Power Equations

---

$$E = C_L V_{DD}^2 P_{0 \rightarrow 1} + t_{sc} V_{DD} I_{peak} P_{0 \rightarrow 1} + V_{DD} I_{leakage}$$

$$f_{0 \rightarrow 1} = P_{0 \rightarrow 1} * f_{clock}$$

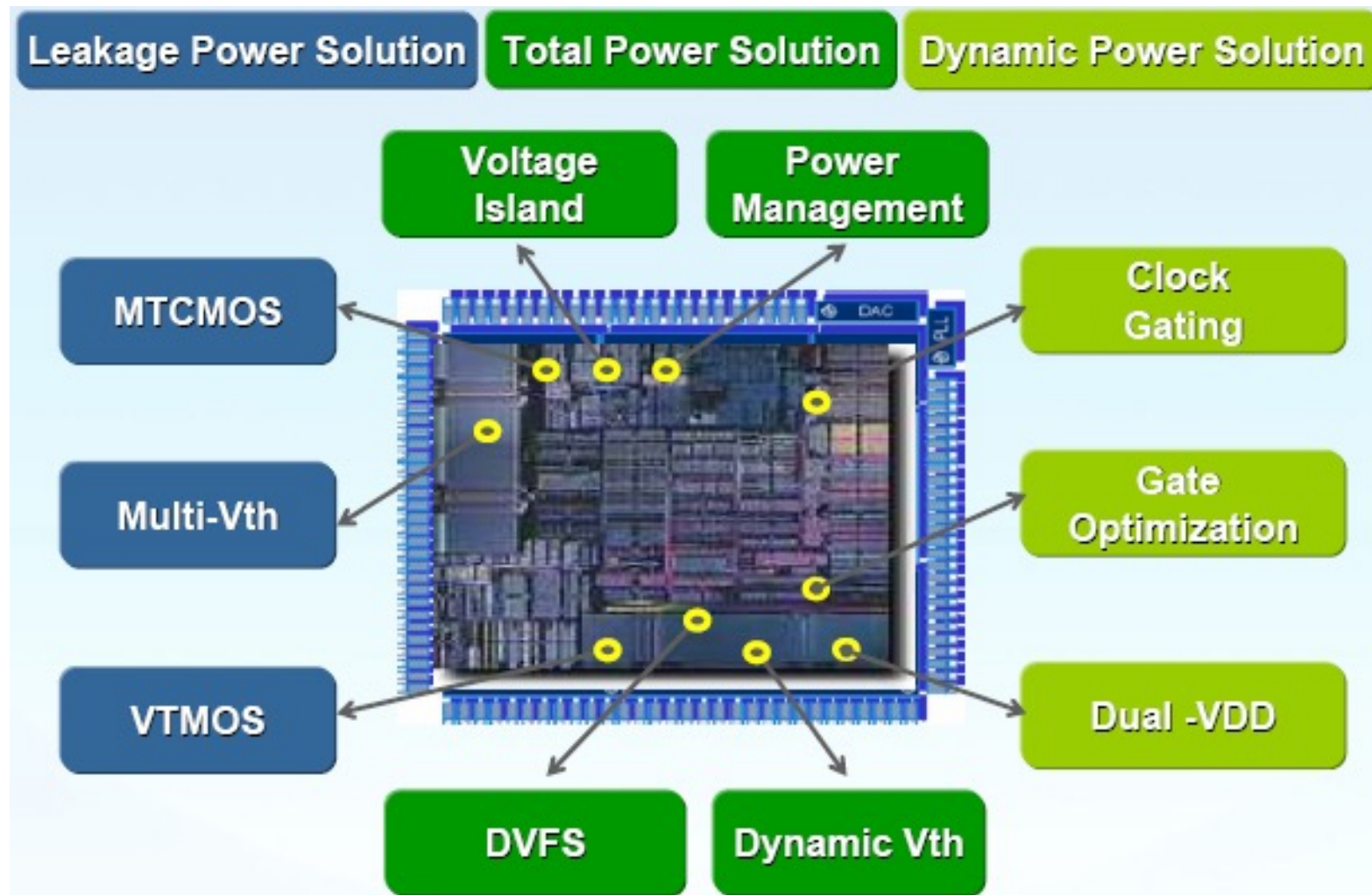
$$P = C_L V_{DD}^2 f_{0 \rightarrow 1} + t_{sc} V_{DD} I_{peak} f_{0 \rightarrow 1} + V_{DD} I_{leakage}$$

Dynamic power

Short-circuit  
power

Leakage power

# Low Power Design Methodologies



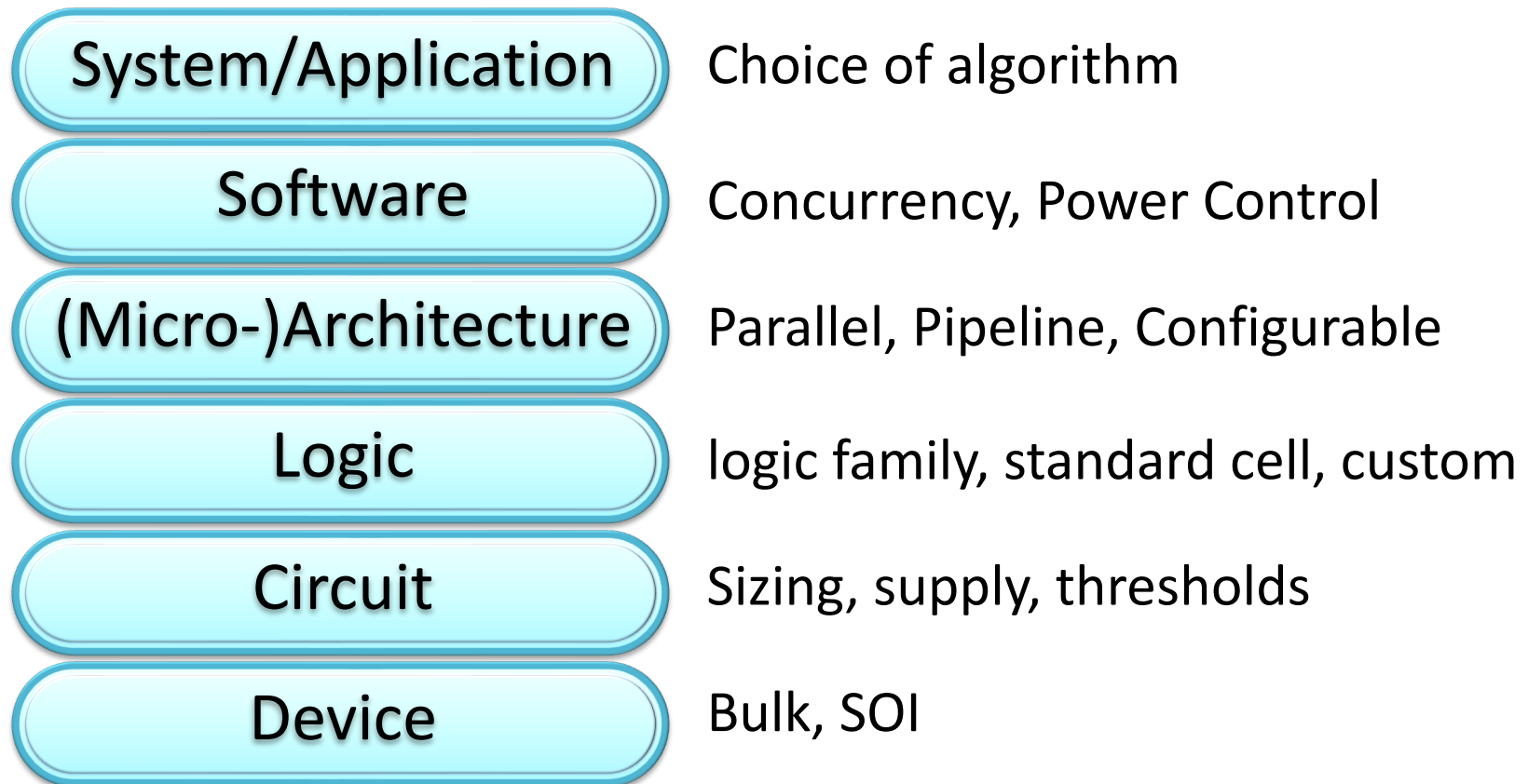
# Power and Energy Design Space

	Constant Throughput/Latency	Variable Throughput/Latency	
Energy	Design Time	Non-active Modules	Run Time
Active	Parallelism, pipeline Reduced switching activity Scaled $V_{dd}$ Transistor sizing Multi- $V_{dd}$	Clock Gating	DFS, DVS, DFVS (Dynamic Freq, Voltage Scaling) (PWM)
Leakage	Transistor stacking + Multi- $V_T$	Power Gating Multi- $V_{dd}$ Variable $V_T$	+ Variable $V_T$

# Design Abstraction Stack

---

- A very rich set of design parameters to consider!  
It helps to consider options in relation to their abstraction layer.



# Active Power Reduction - Switching

---

$$P = \alpha \cdot f_{clk} \cdot C_L \cdot V^2$$

## ■ Reduce switching activity

- ◆ Conditional execution
- ◆ Data gating
- ◆ Clock gating
- ◆ Glitch reduction
- ◆ Conditional precharge for dynamic circuits
- ◆ Turn off inactive blocks
- ◆ Reduce toggling of high capacitance nodes buses

# Active Power Reduction - Frequency

---

$$P = \alpha \cdot f_{clk} \cdot C_L \cdot V^2$$

- Frequency reduction with the same performance
  - ◆ Use parallelism
  - ◆ Time borrowing technique
  - ◆ Pipeline retiming technique
  - ◆ Less pipeline stages
  - ◆ Use double-edge flip-flops
  - ◆ Multi-clock domain
  - ◆ Asynchronous circuits
  - ◆ Globally asynchronous locally synchronous (GALS)
  - ◆ Self-timed circuits

# Active Power Reduction - Capacitance

---

$$P = \alpha \cdot f_{clk} \cdot C_L \cdot V^2$$

## ■ Reduce switching capacitance

- ◆ Minimize diffusion, wire and gate loading
- ◆ Minimize loading in high activity factor nodes (clocks, dynamic circuits)
- ◆ Coupling-aware wire routing
- ◆ Use more efficient layout technique
- ◆ Buffer insertion
- ◆ Reduce long wires



# Active Power Reduction - Voltage

---

$$P = \alpha \cdot f_{clk} \cdot C_L \cdot V^2$$

- Supply voltage scaling is slowing down
- Thresholds don't scale
- Voltage reduction
  - ◆ Technology scaling
  - ◆ Dynamic voltage scaling (DVS)
  - ◆ Multi-Vdd Design
  - ◆ Power domain
  - ◆ Low-voltage design
  - ◆ On-chip integrated voltage regulation module (VRM)

# Principles for Active Power Reduction

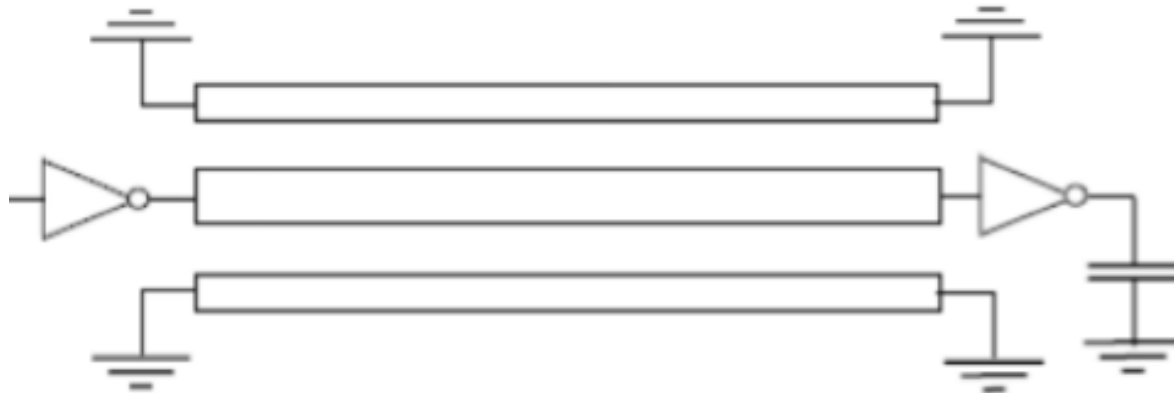
---

- Prime choice : Reduce voltage!
  - ◆ Recent years have seen an acceleration in supply voltage reduction
  - ◆ Design at very low voltages still open question
- Reduce switching activity
- Reduce physical capacitance

# Design Techniques for Coupling Effect

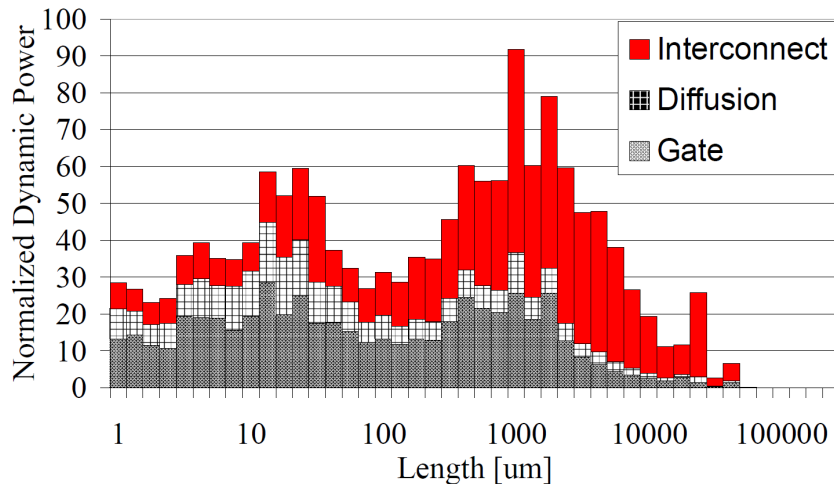
---

- Widen the pitch between adjacent lines
- Routing wires by different metal layers
- Change the geometrical shape of interconnects
- Bus coding schemes
- Phase coding schemes
- Add shielding lines



# Power of Interconnects

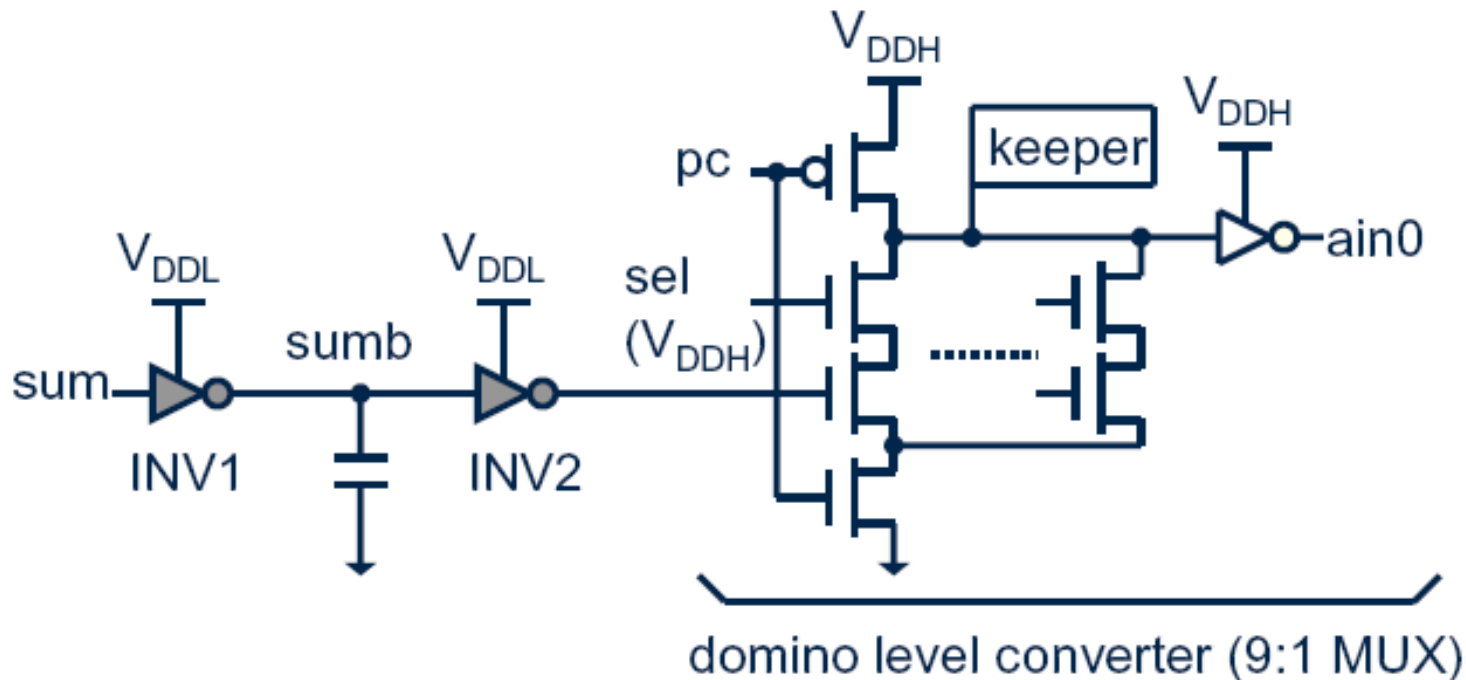
- Interconnect consumes >50% of dynamic power in a micro-processor
- 90% of interconnect power is in 10% of interconnect



Low Power Interconnect	BW (Ghz)	Swing (V)	Normalized Energy
Basic ( no scaling)	>1	1	1
Low swing (Single-ended )	<0.25	0.6	0.6
Differential Pair	>1	0.05	0.8
Capacitive	<0.25	0.05	0.2

# Low Swing Bus & Level Converter

- Delay of INV1 does not increase
- INV2 is placed near 9:1 MUX to increase noise immunity
- Level conversion is done by a domino 9:1 MUX



# **Clock Domains and Clock Gating**

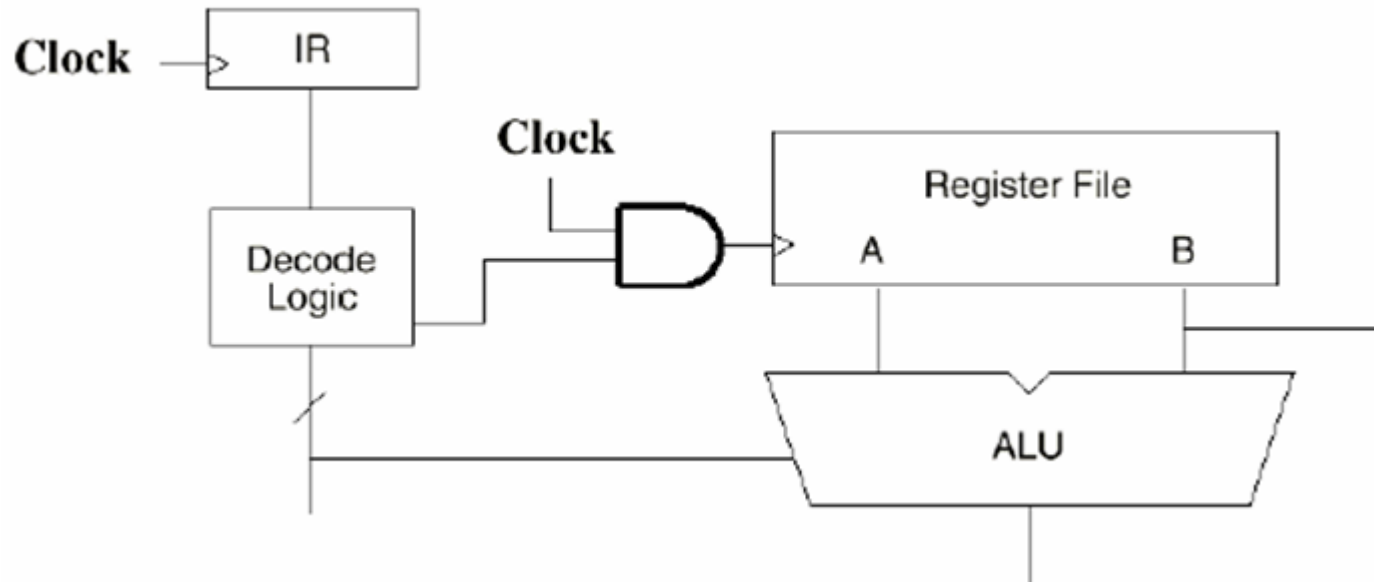
---

- Clock distribution & clock gating reduce active power
- **Clock Domains**
  - ◆ On-chip clock distribution generates multiple synchronous phase aligned clock domains
- **Clock Gating**
  - ◆ **IP Core Level** clock gating disables clocks to whole IP blocks that are not currently being used
  - ◆ **Register Level** clock gating disables clock to unused portions of IP blocks during each operation or instruction cycle

# Clock Gating

---

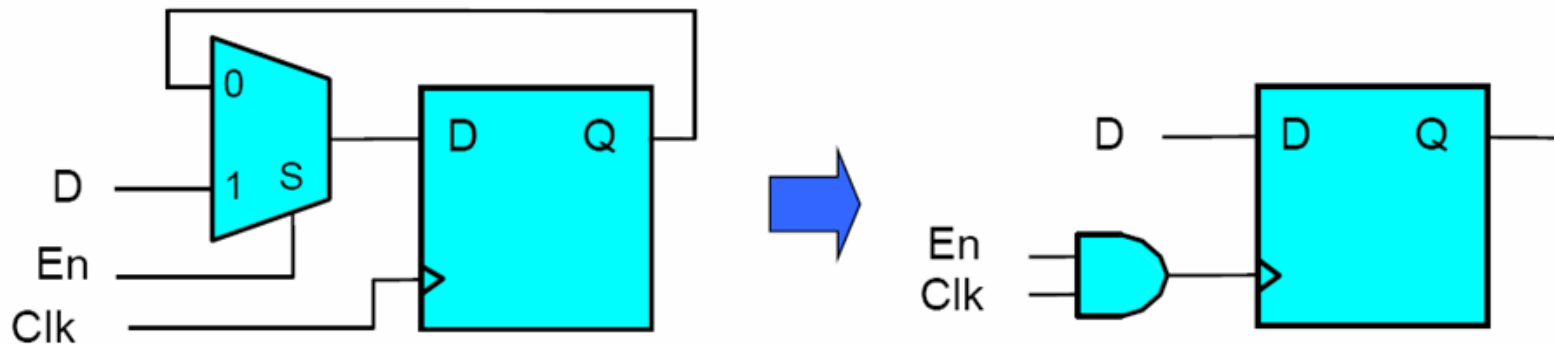
- Requires careful skew control
  - ◆ Well handled in today's EDA tool



# Clock Gating Design

---

- Save power by gating the clock when data activity is low
- Widest used switching power reduction technique
- Requires early EN signal arrival, as well as detailed timing and logic validation

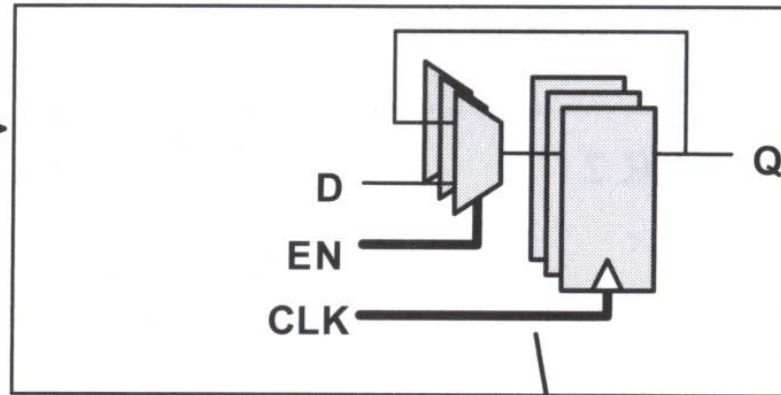




# Clock Gating in Verilog



Typical  
compile

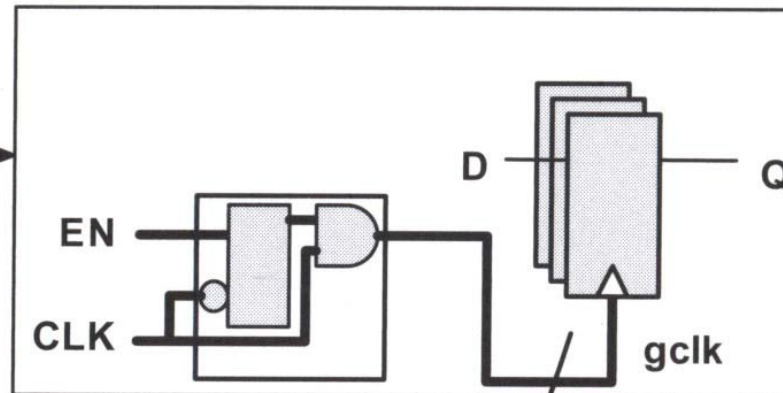


High  
activity

```
always@ (posedge CLK)
  if (EN)
    Q <= D;
```



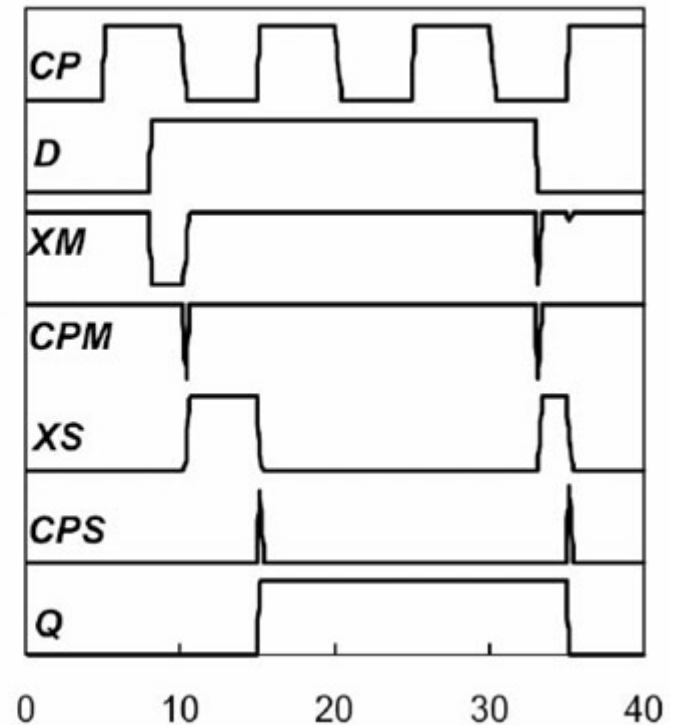
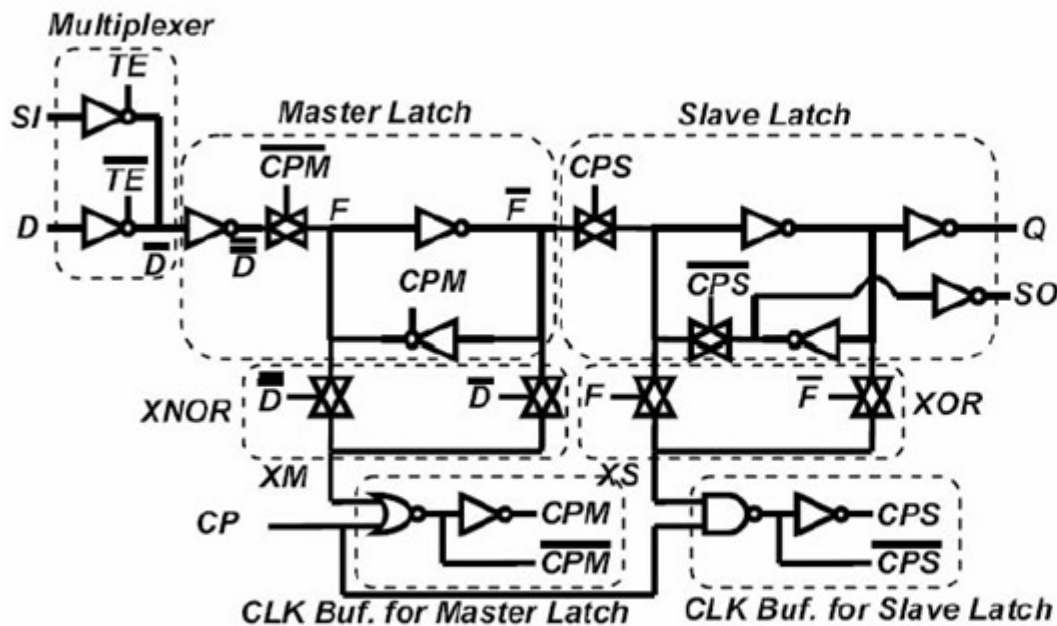
Compile with  
clock gating  
insertion



Low  
activity

# Conditional Clocking Flip-Flop

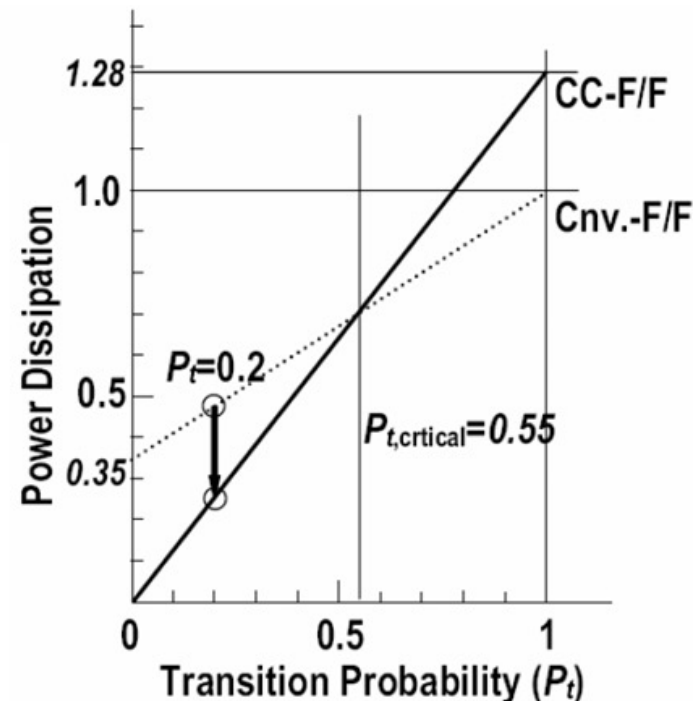
- FF does not consume active power when the data input does not change its state



# Power Comparisons of CCFF

- Taking into account the overhead of the auxiliary circuits, the flip-flop consumes less power than the conventional flip-flops when the data transition probability is less than 55%.
- Issues: leakage, setup time

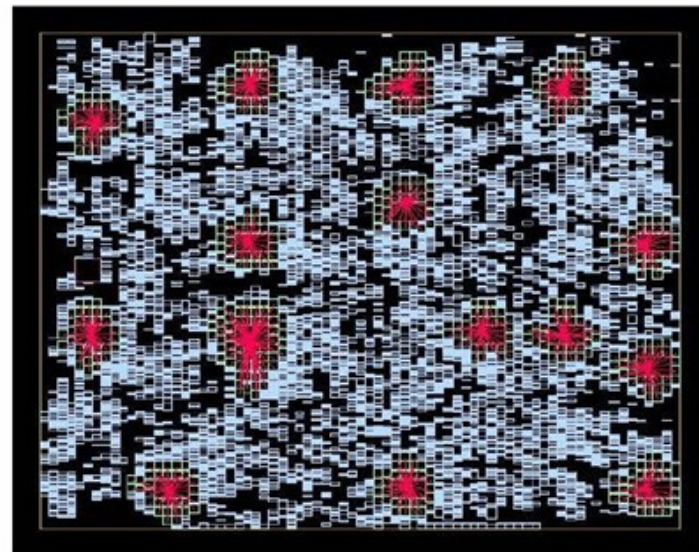
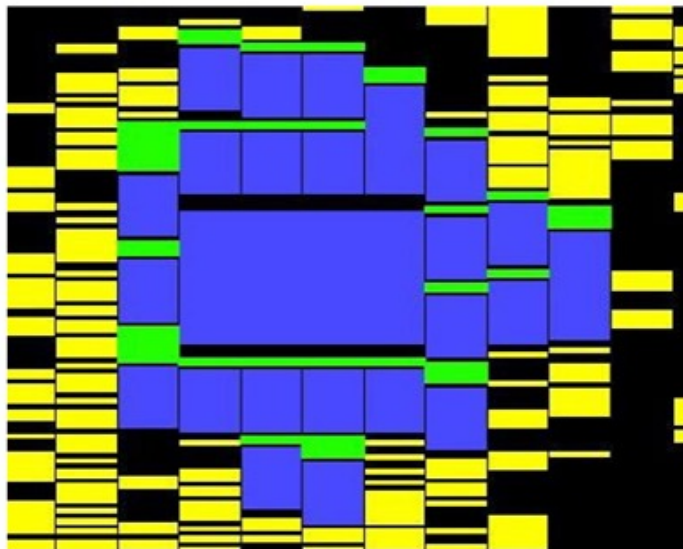
		conventional	conditional clk
Power	$P_{LHHL}$	1.00	0.35
	$P_{LLHH}$	1.28	0.00
Delay (ps)	$CP$ -to- $Q$	82	86
	Setup	84	199
	Hold	-72	-195
Area		1.00	1.33



# Latch Clustering

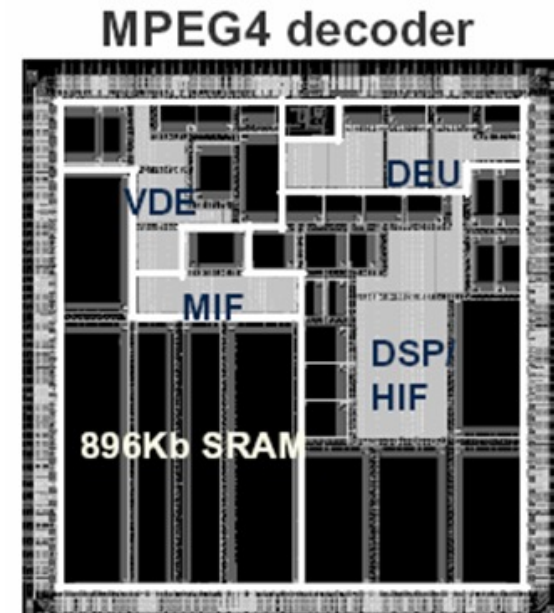
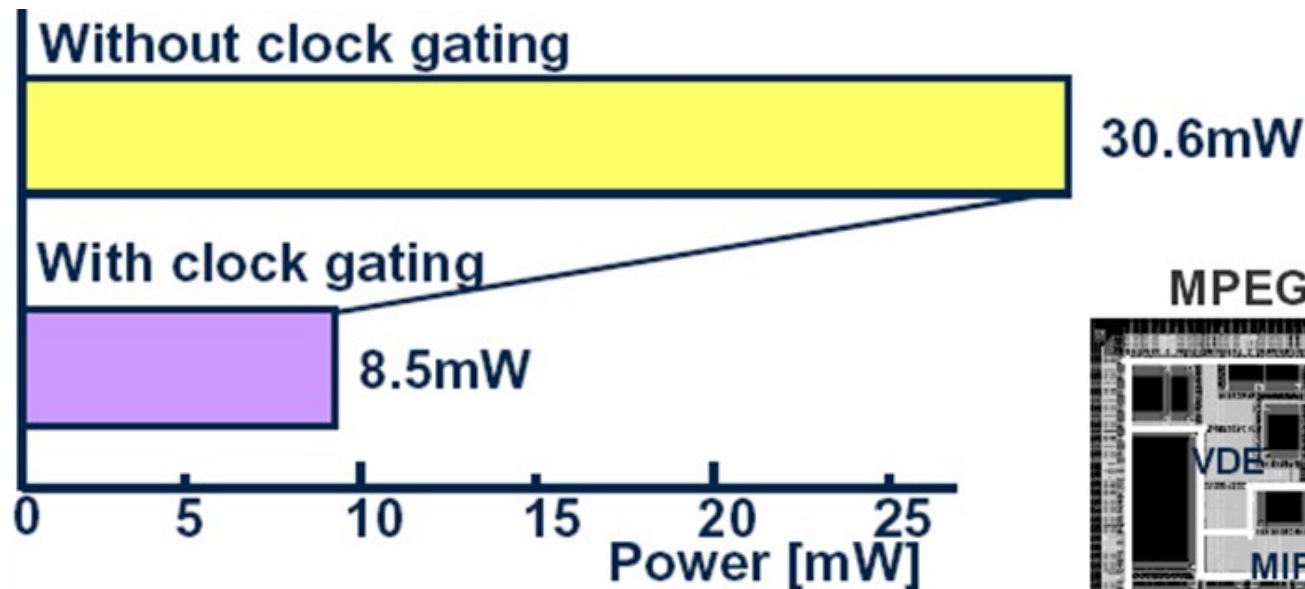
---

- Minimize the capacitive loading on local clock buffers by clustering latches around them
- ◆ Tradeoff between latch placement flexibility and clock power saving
- ◆ Reduction in clock skew between capturing and launching latch compensates for loss in latch placement flexibility



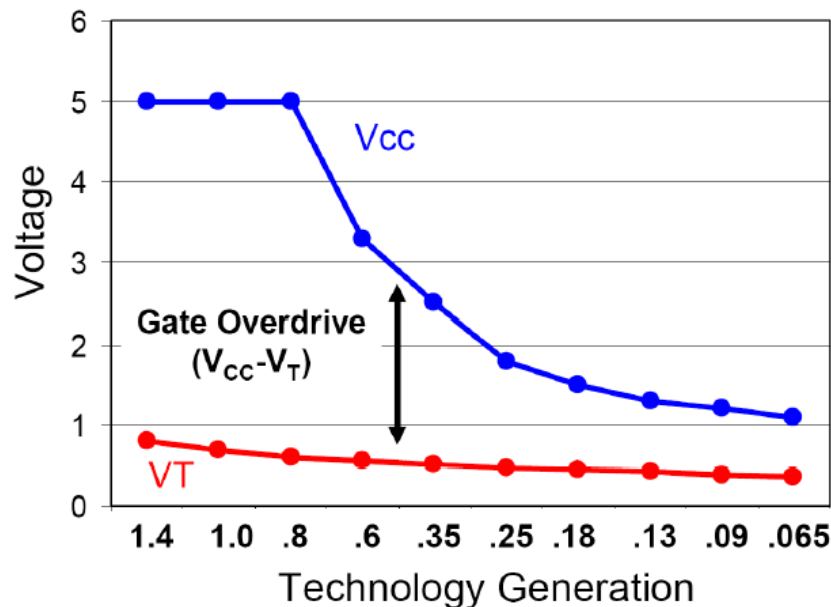
# Power Saving of Clock gating

- 90% of flip-flops were gated.
- 70% power reduction by clock-gating alone



# Voltage Scaling Trends

- Vcc scaling has been driven by power and oxide reliability
- Gate overdrive is decreasing with each technology generation
- $V_T$  is scaling very slowly
- Vcc scaling trend is decreasing due to performance concerns



# Controlling VDD and VTH

	Active	Stand-by
Multiple $V_{TH}$	Dual- $V_{TH}$	MTCMOS
Variable $V_{TH}$	$V_{TH}$ hopping	VTCMOS
Multiple $V_{DD}$	Dual- $V_{DD}$	Boosted gate MOS
Variable $V_{DD}$	$V_{DD}$ hopping	

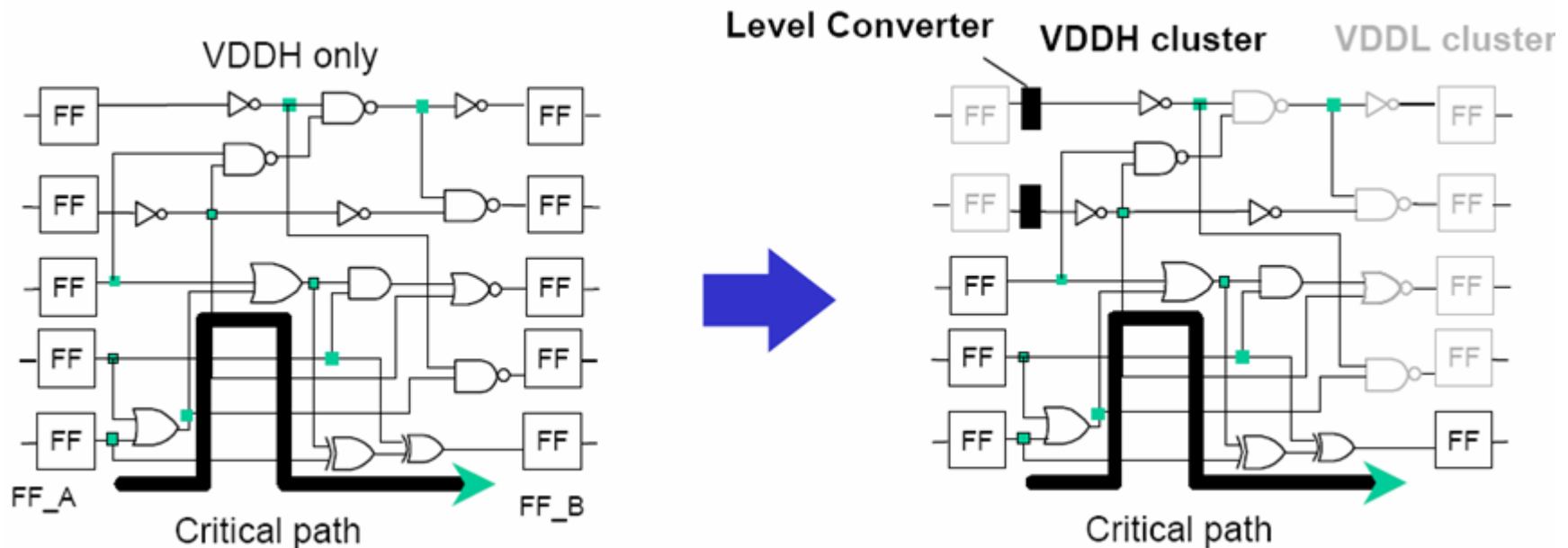
Software-hardware cooperation

Technology-circuit cooperation

- MTCMOS : Multi-Threshold CMOS
- VTCMOS : Variable Threshold CMOS
  - ◆ Multiple : spatial assignment
  - ◆ Variable : temporal assignment

# Cell-Level Dual-VDD Approach

- Use reduced voltage VDDL in non-critical paths
- Apply original voltage VDDH to timing critical path

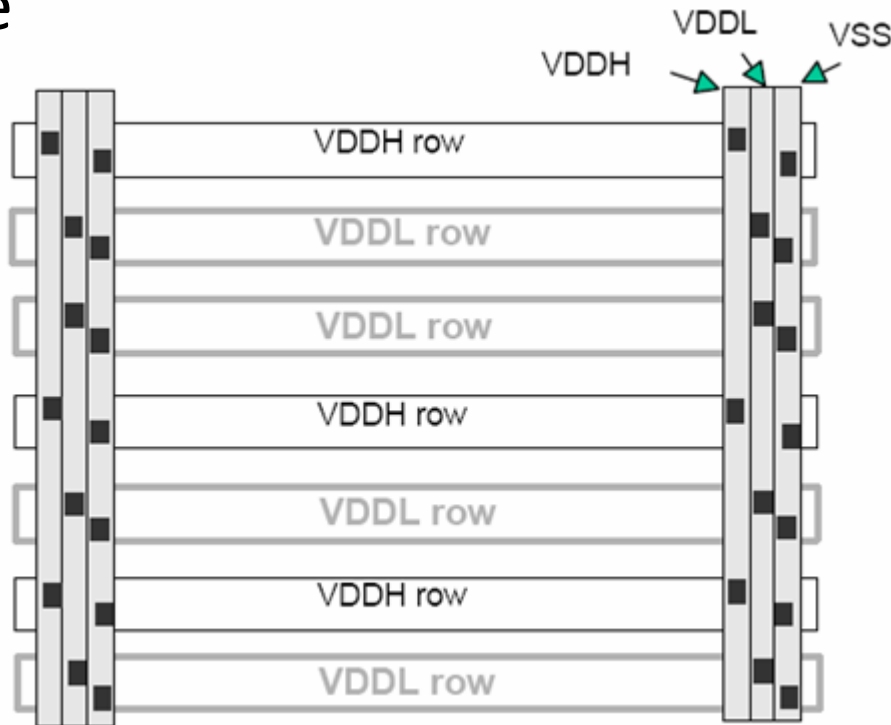


**Challenges: minimize # of level converters by clustering**



# Row-Based Cell-Level Dual-VDD

- P&R tool determines which row should be VDDL
- Clock tree synthesis using VDDL clock buffers
- 25% power reduction demonstrated on H.264 video codec core



**Row-by-row layout architecture with Dual-Vdd**

# Multiple Supply Voltages

---

## ■ Multiple Supplies in a Block

- ◆ Only the critical path cells work on Higher Vdd.
- ◆ Level conversion at different places within the block.
- ◆ Implementation and Physical Design Challenges. Not trivial with standard-cell based design.

## ■ Block Level Supply Assignment

- ◆ “Voltage Islands”
- ◆ Higher throughput/Lower latency functions are implemented in higher Vdd.
- ◆ Separate Supply distribution grids, Level conversion performed at Block Boundaries.

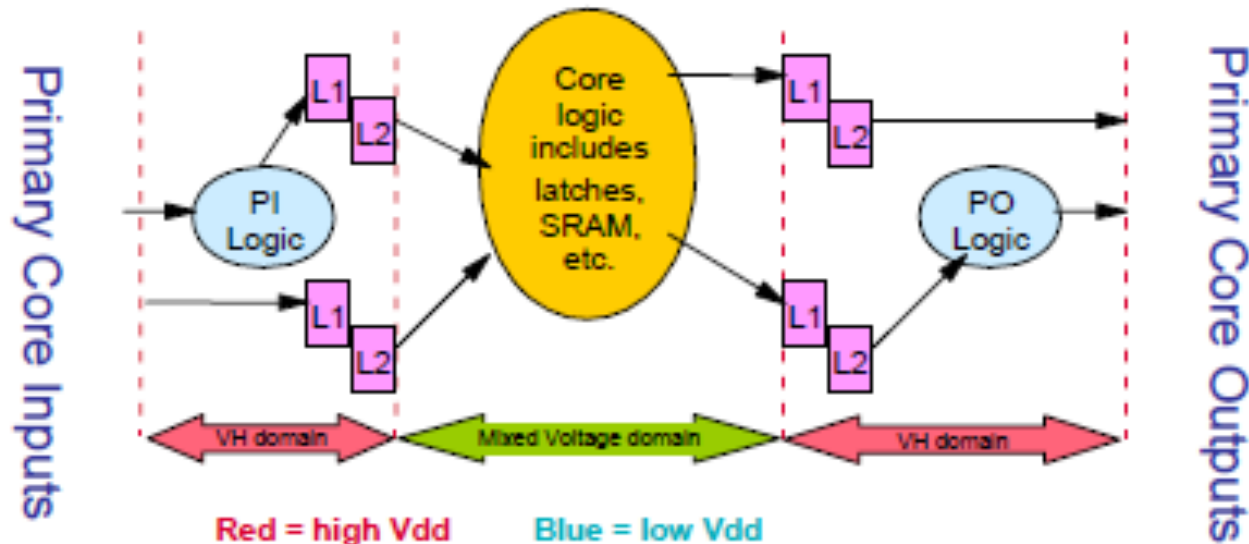
# Voltage Islands

## ■ What's Voltage Islands ?

- ◆ Regions supplied through separate, dedicated power feeds

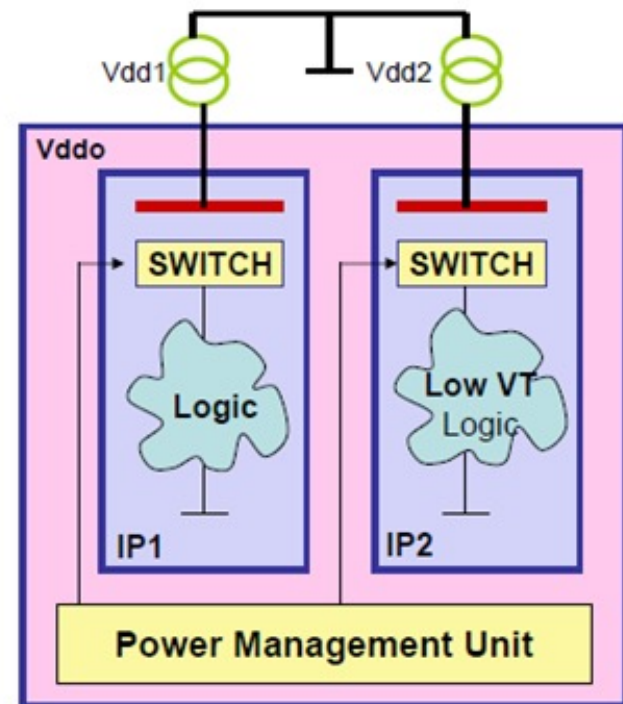
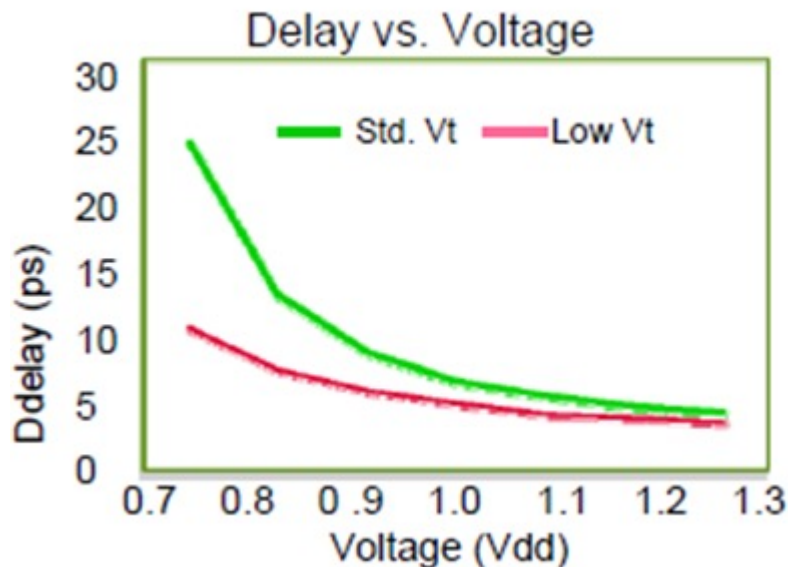
## ■ Power Saving through Reduced Voltage

- ◆ Goal is to define groupings of circuit or macros within a system-on-a-chip which can be powered by a lower supply while maintaining the required frequency and offering lower power consumption.

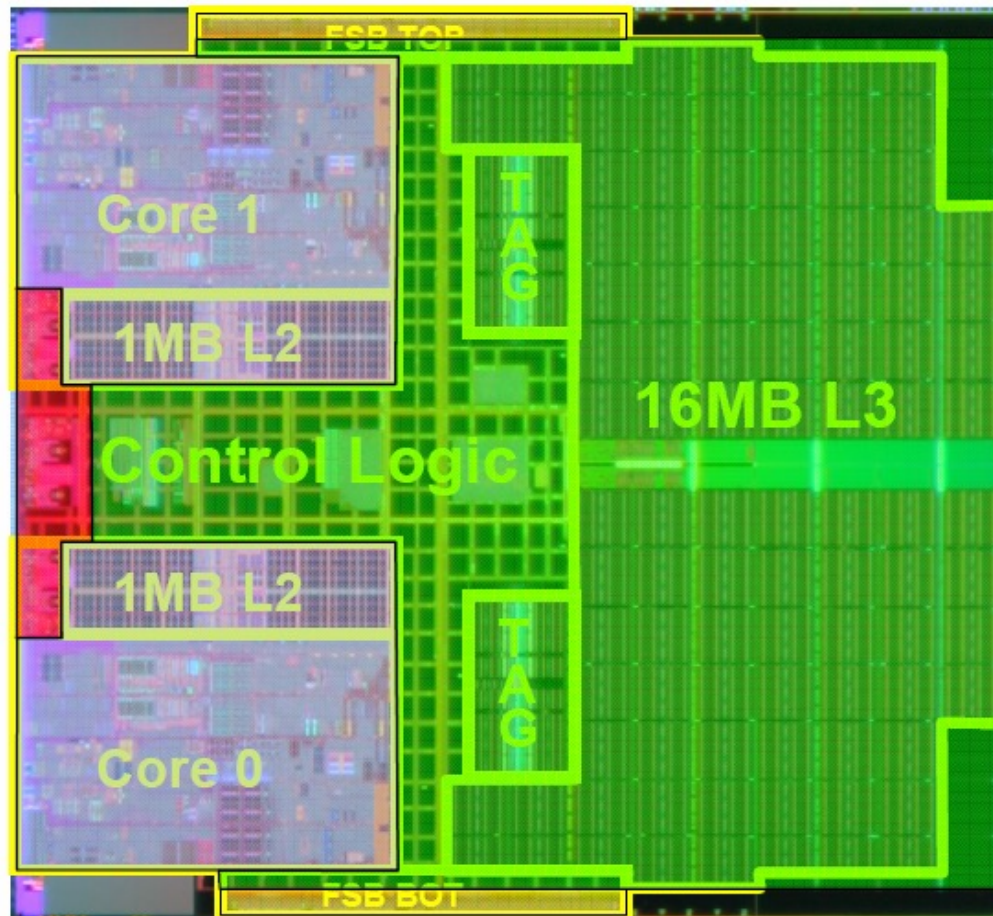


# Voltage Island Concept

- Tradeoff between power and delay by running function blocks at different voltages.
- Can use mix of low and high  $V_t$  to balance performance and leakage
- Switch off inactive blocks to reduce leakage power
- Requires IP standards for power management, clock gating, etc



# Multiple voltage Domain

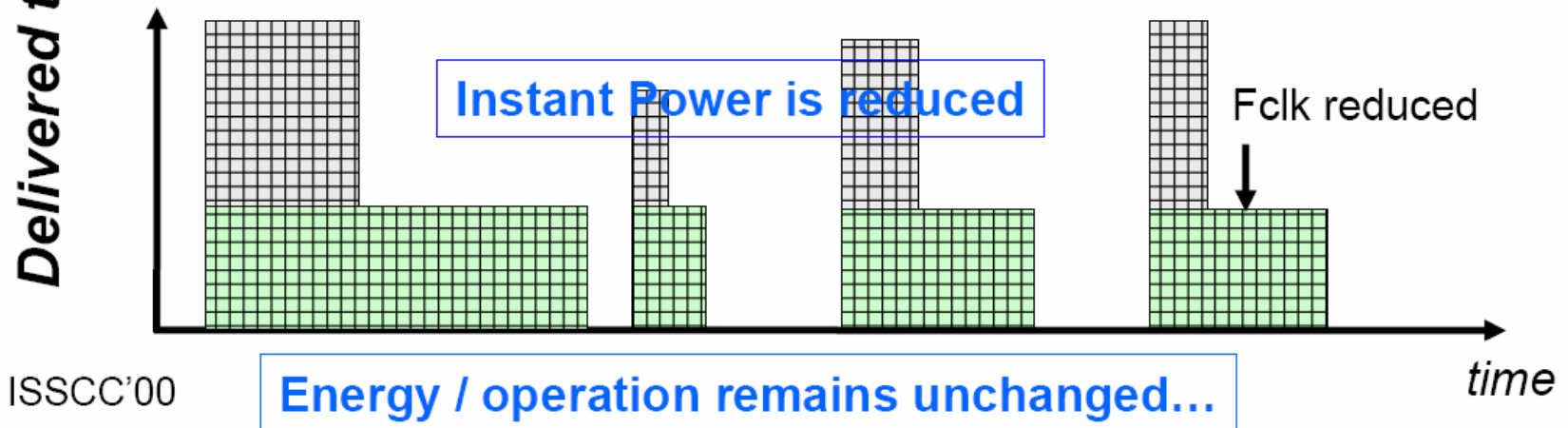
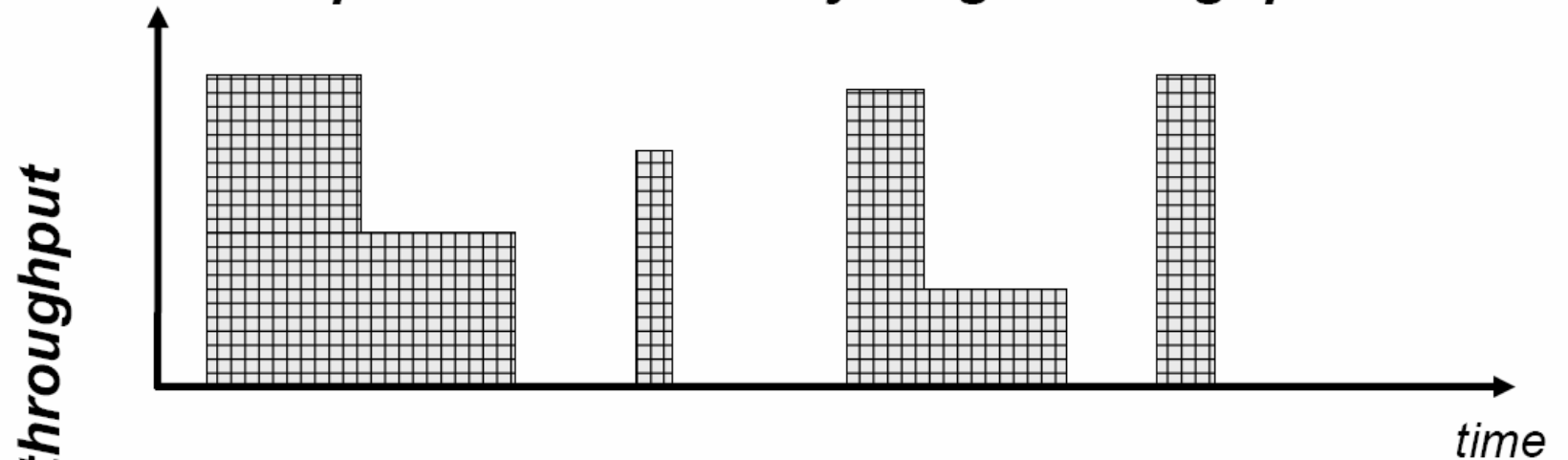


Legend:  Core  PLL  Uncore  I/O

S. Rusu, ISSCC 2006

# Frequency Scaling

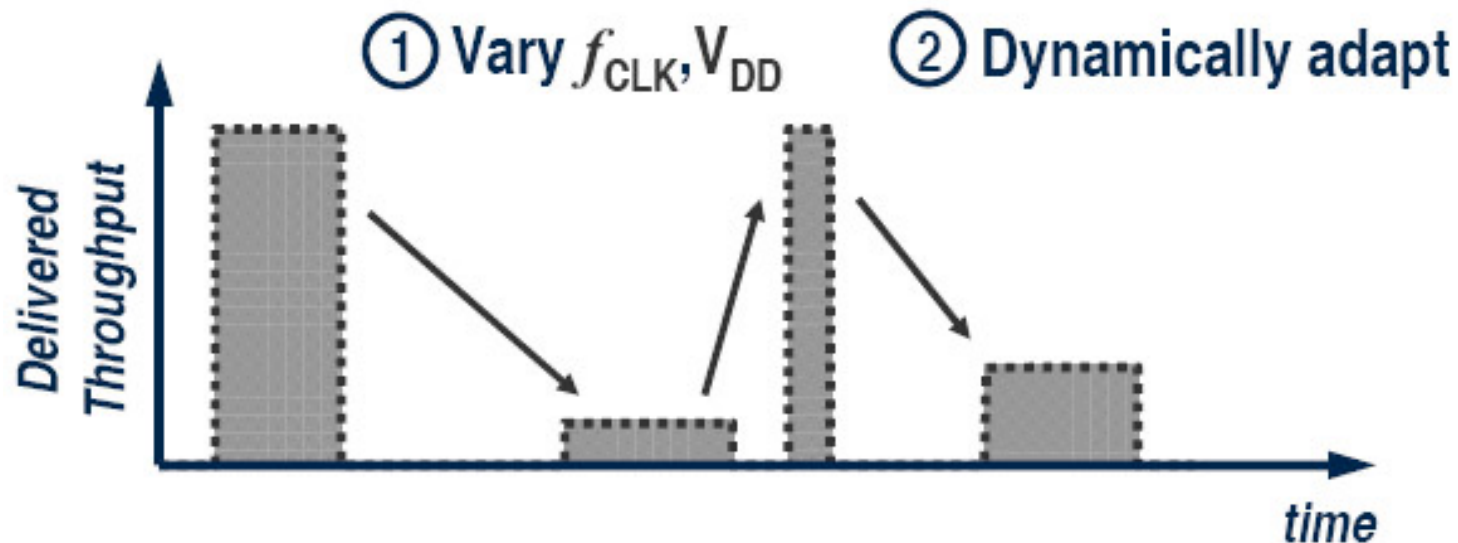
*Compute ASAP = always high throughput*



Burd, ISSCC'00

# Dynamic Voltage Scaling (DVS)

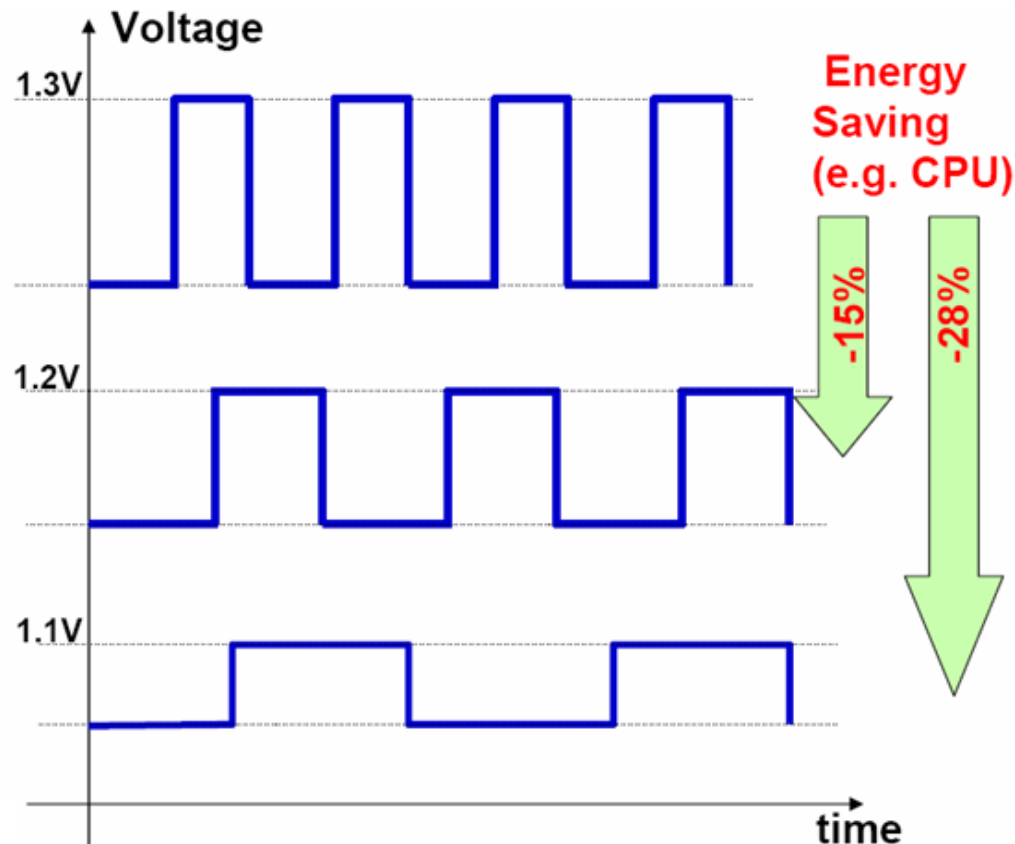
- Dynamically scale energy/operation with throughput.
- Always minimize speed → minimize average energy/operation.
- Extend battery life up to 10x with the exact same hardware!



# Dynamic Voltage & Frequency Scaling

## ■ Process requirements

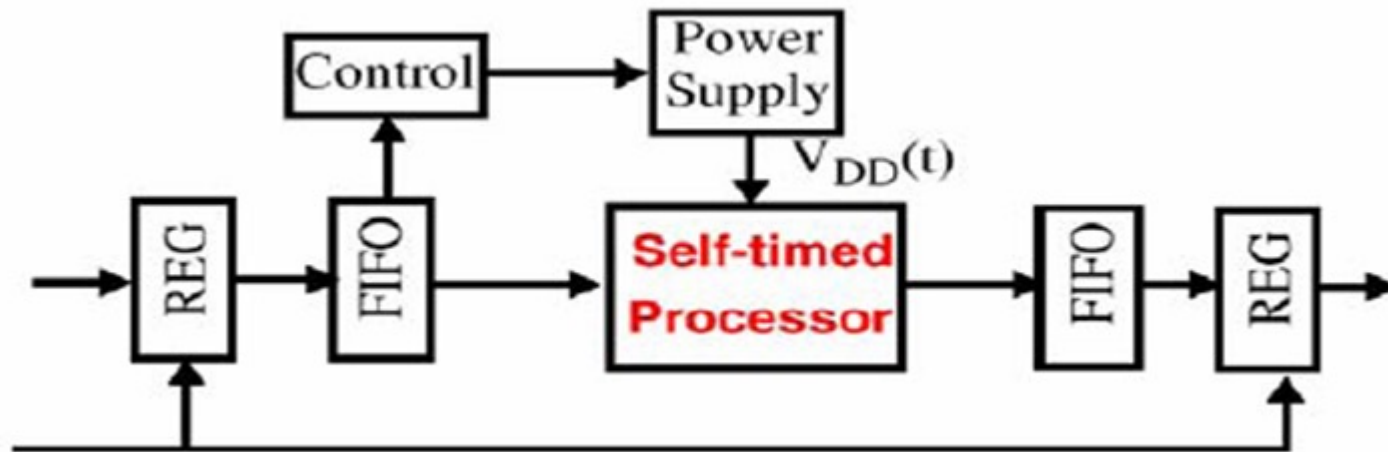
- ◆ Enough voltage excursion & characterization
- ◆ Low leakage level (Tasks take long to execute)



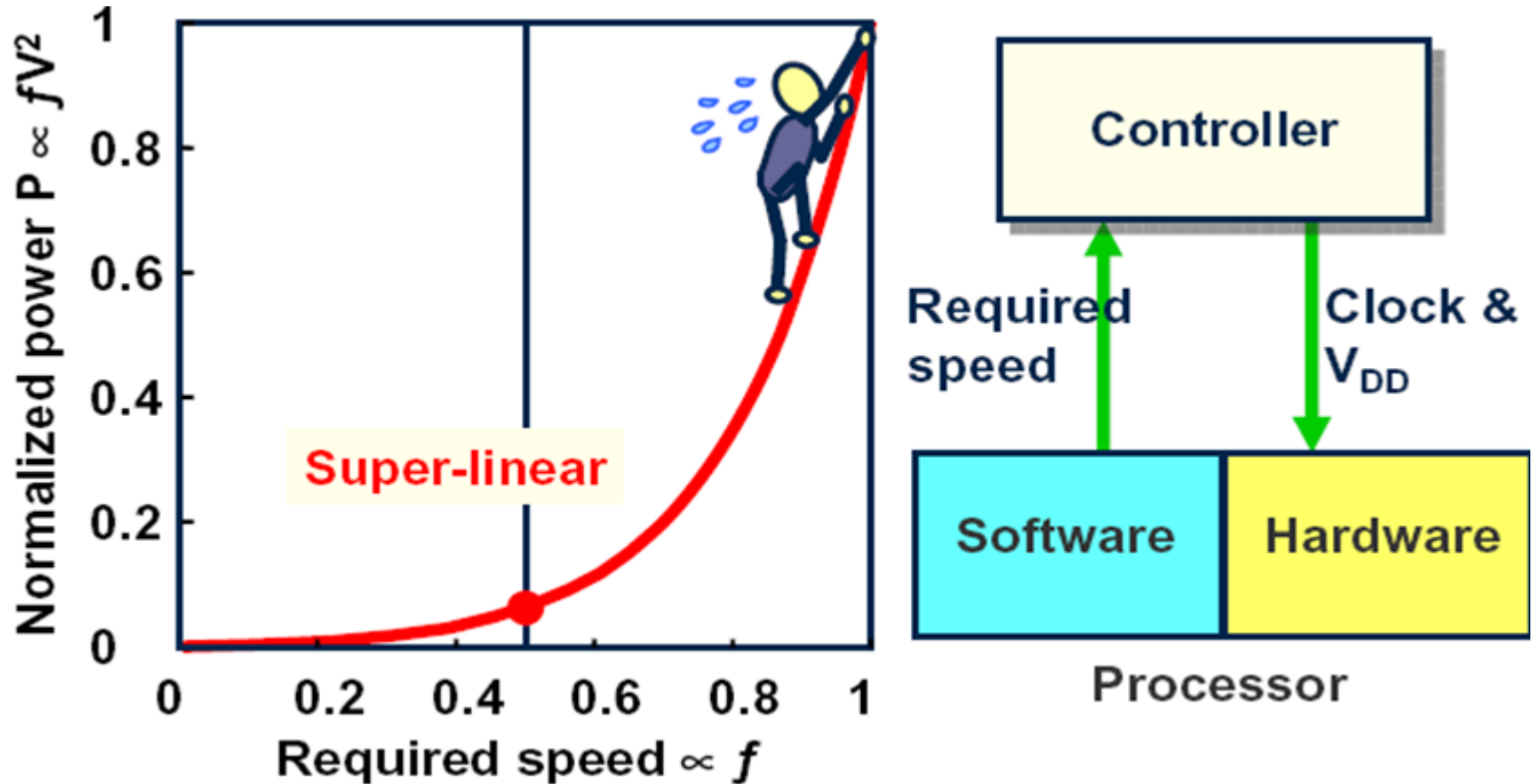


# Adaptive Supply Voltages

- Exploit data dependent computation times to vary the supply



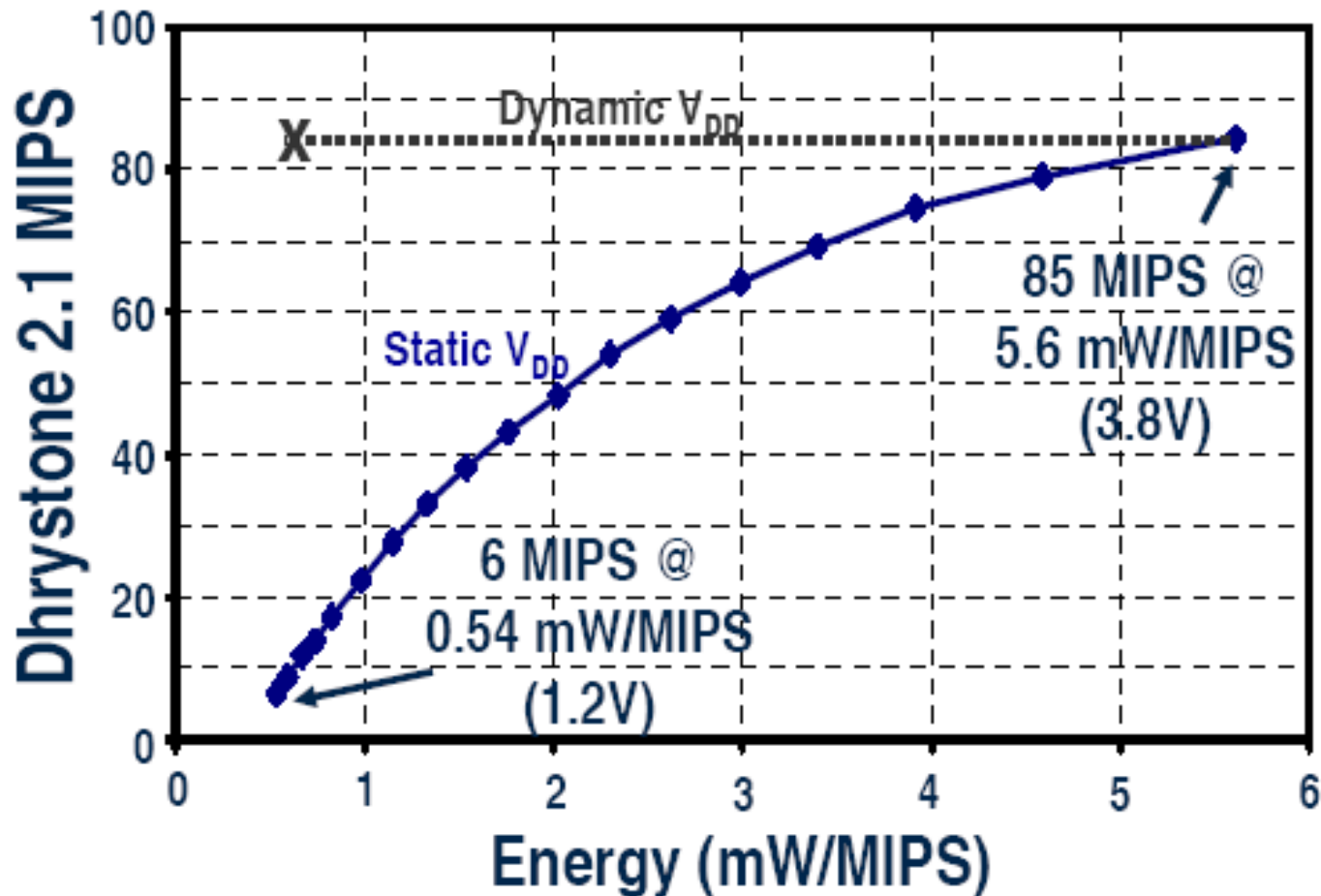
# Software-Hardware Cooperation



If you don't need to hustle, relax and save power.

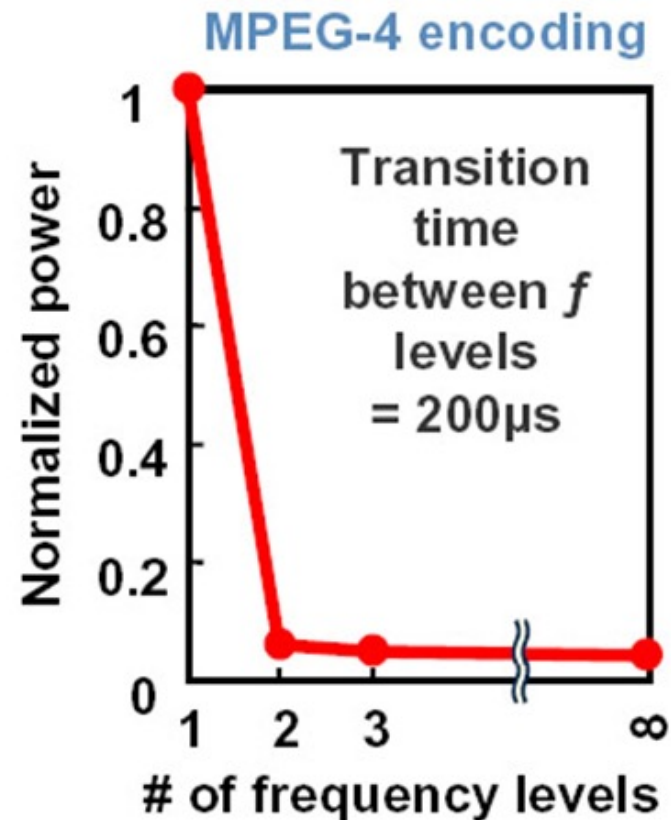
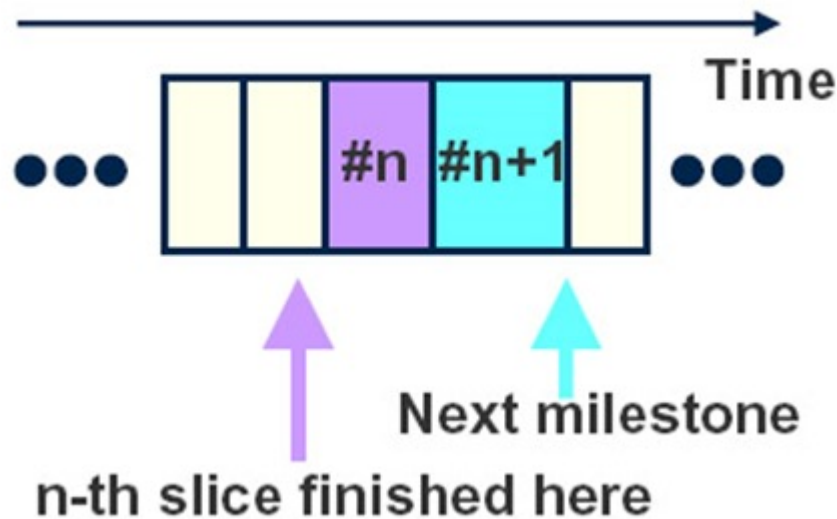
# Measured System Performance & Energy

- Dynamic operation can increase energy efficiency  $> 10\times$ .



# $V_{DD}$ -Hopping

- Application slicing and software feedback guarantee real-time operations.
- Two hopping level are sufficient.



# **Challenge : Design over Wide Range of Voltages**

---

- Circuit design constraints. (Functional verification)
- Circuit delay variation. (Timing verification)
- Noise margin reduction. (Power grid, coupling)
- Delay sensitivity. (Local power distribution)

**Design verification complexity similar to high-performance processor design @ fixed VDD**

# Multi-Mode Multi-Corner

---

- Corner: defined as a set of libraries characterized for process, voltage and temperature variations.
  - ◆ Corners are not dependent on functional settings
  - ◆ To capture variations in the manufacturing process, along with expected variations in the ***voltage and temperature*** of the environment in which the design will operate.
- Mode: defined by a unique set of clocks, supply voltages, and timing constraints in similar operating conditions.
  - ◆ It can also have annotation data, such as SDF or parasitic files.

# MCMM (or MMMC) Optimization

---

- MCMM optimization is useful for designs that can operate in many modes such as test mode, low-power active mode, stand-by mode and so on.
- Used along with specification of power intent in the **Unified Power Format (UPF)**, it serves as the key enabling technology for performing dynamic voltage and frequency scaling (DVFS) design realization

# Timing Margins

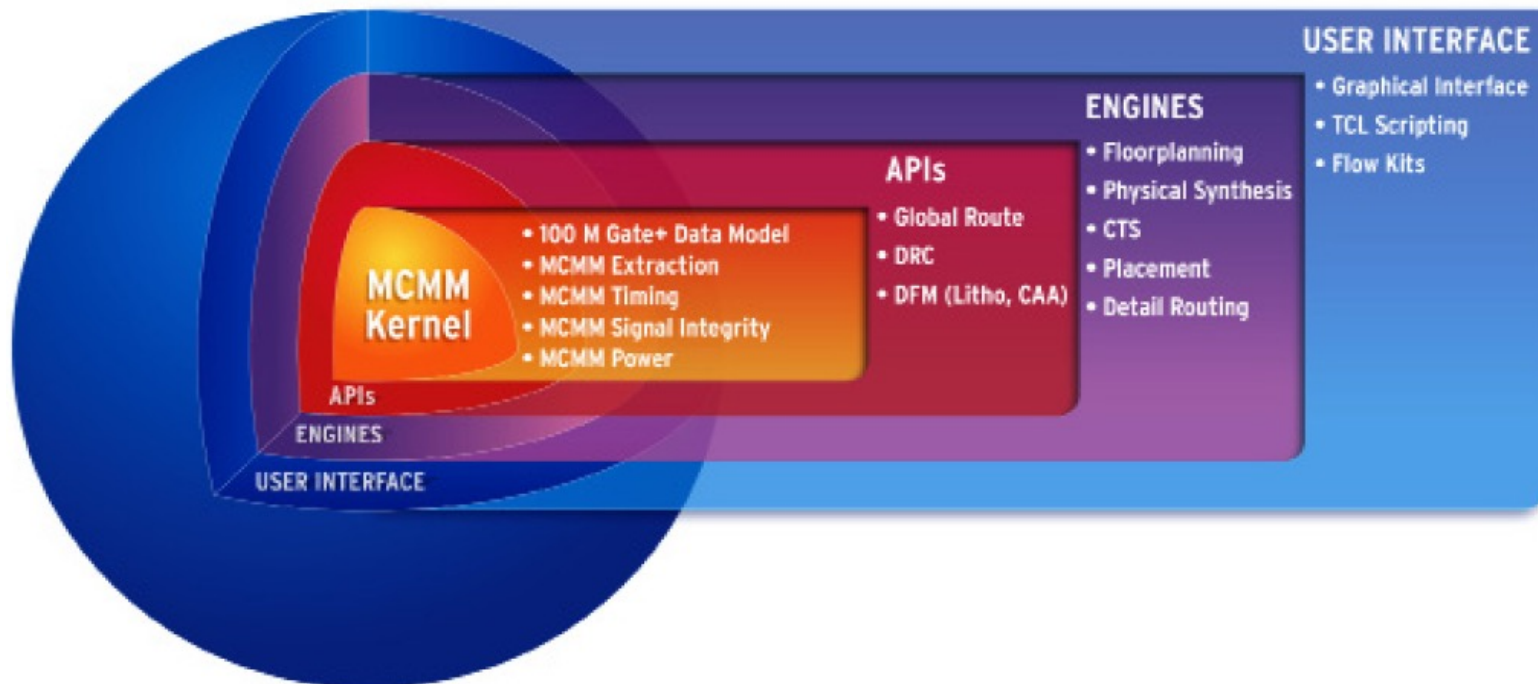
- The setup and hold times must be analyzed simultaneously for different combinations of library models, voltages, and interconnect (RC) corners.

	Single Core Design			Core + 1 Island				Core + 2 Islands				
	Lib	Core	RC	Lib	Core	Vdd1	RC	Lib	Core	Vdd1	Vdd2	RC
Setup1	Max	1.2	Max	Max	1.2	0.9	Max	Max	1.2	0.9	0.9	Max
Setup2	Max	1.2	Min	Max	1.2	0.9	Min	Max	1.2	0.9	0.9	Min
Hold1	Min	1.8	Min	Min	1.8	1.5	Min	Min	1.8	1.5	1.5	Min
Hold2	Min	1.8	Max	Min	1.8	1.5	Max	Min	1.8	1.5	1.5	Max
Setup1	—	—	—	Max	1.2	0	Max	Max	1.2	0	1.2	Max
Setup2	—	—	—	Max	1.2	0	Min	Max	1.2	0	1.2	Min
Hold1	—	—	—	Min	1.8	0	Min	Min	1.8	0	1.8	Min
Hold2	—	—	—	Min	1.8	0	Max	Min	1.8	0	1.8	Max
Setup1	—	—	—	—	—	—	—	Max	1.2	0.9	1.2	Max
Setup2	—	—	—	—	—	—	—	Max	1.2	0.9	1.2	Min
Hold1	—	—	—	—	—	—	—	Min	1.8	1.5	1.8	Min
Hold2	—	—	—	—	—	—	—	Min	1.8	1.5	1.8	Max
Setup1	—	—	—	—	—	—	—	Max	1.2	0	0.9	Max
Setup2	—	—	—	—	—	—	—	Max	1.2	0	0.9	Min
Hold1	—	—	—	—	—	—	—	Min	1.8	0	1.5	Min
Hold2	—	—	—	—	—	—	—	Min	1.8	0	1.5	Max



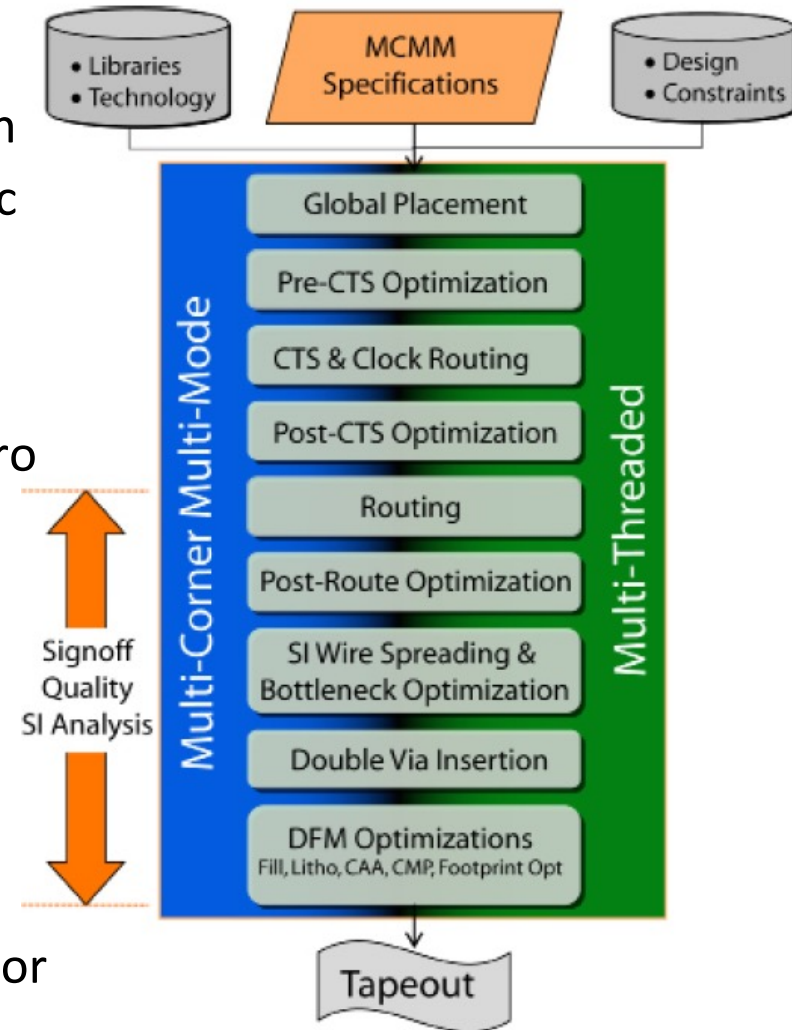
# Example: Mentor Olympus-SoC

- Variation-Based Timing Closure
- MCMM Clock Tree Synthesis
- MCMM Signal Integrity Closure
- Routing for Manufacturability and Yield
- High-Capacity Architecture



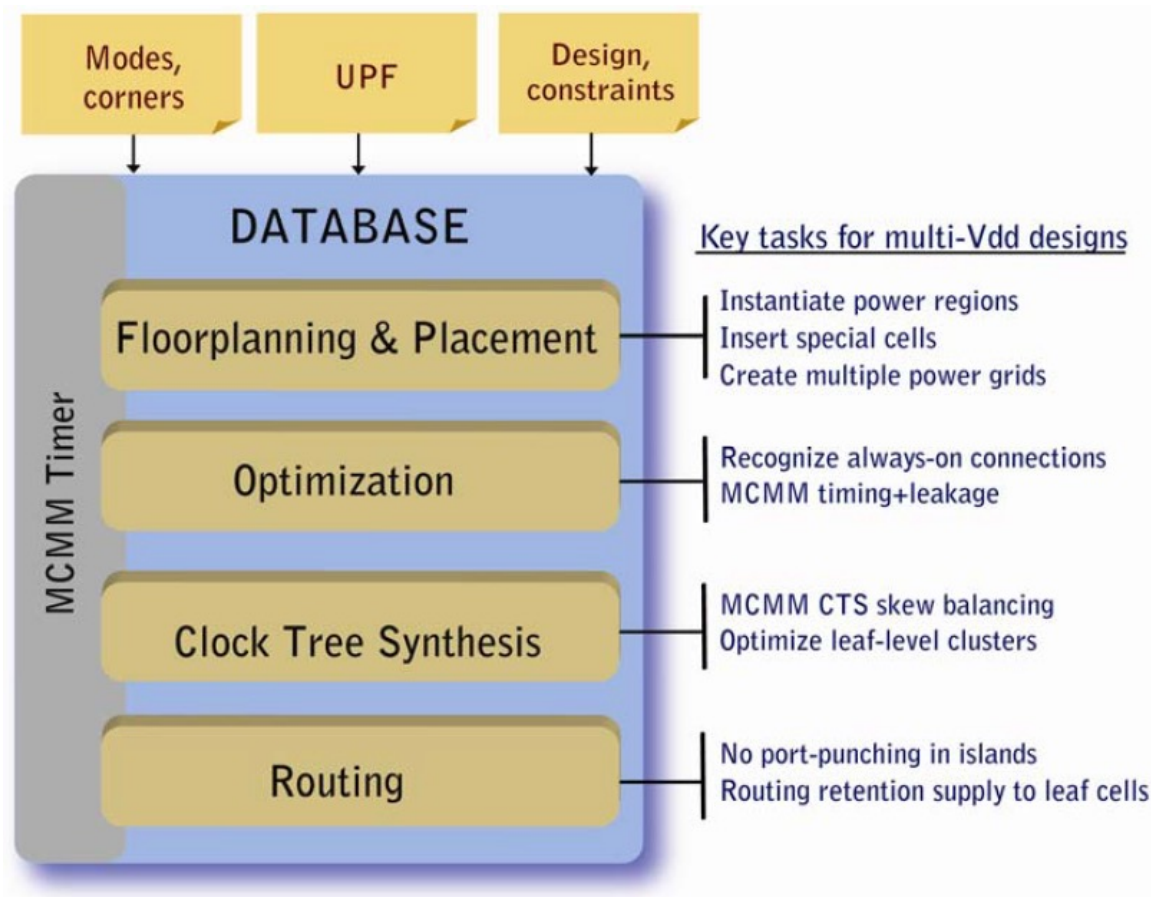
# Features of MCMM Optimization

- Patented MCMM optimization during all steps
- Fast routing with full 40/28 nm rule support
- Sign-off quality timing analysis and optimization
- Extremely fast and accurate, on-the-fly parasitic extraction
- Floorplanning, rapid design feasibility and constraint debugging
- Best-in-class, CTS-aware standard cell and macro placement
- MCMM CTS for robust, low-power clock trees
- MCMM SI to concurrently compute delay shift and glitch for any number of mode/corner scenarios in a single pass
- Advanced physical synthesis with built-in OCV and CPPR
- Handles multi-million gate designs hierarchical or flat with faster runtimes



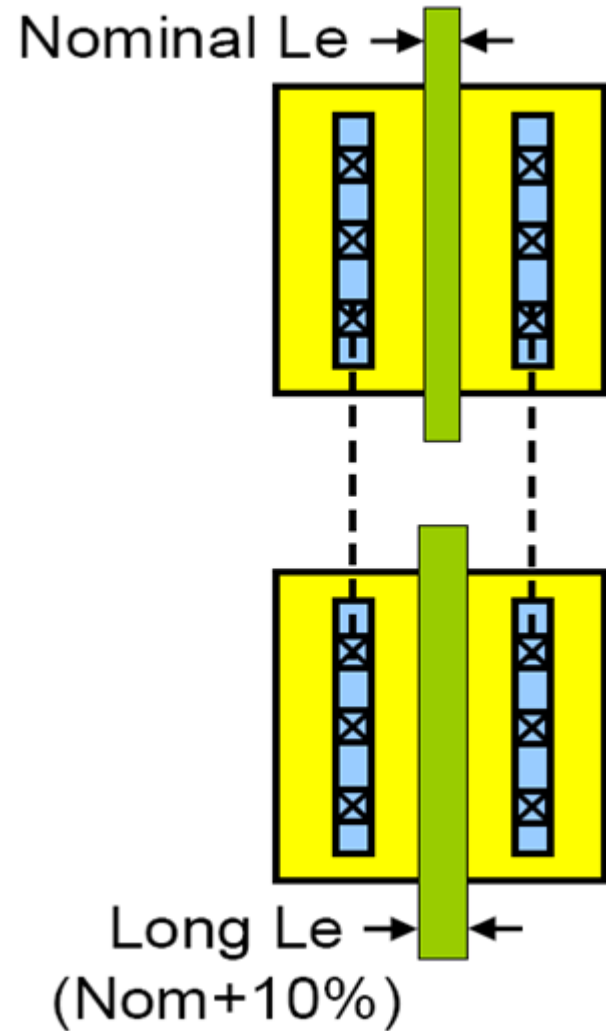
# Physical design flow for multi-voltage designs

■ Power-aware CTS with smart clock gate placement, slew shaping, register clumping, and concurrent MCMM optimization that ensures a balanced clock tree with the minimum number of clock buffers.



# Long-Le Transistors

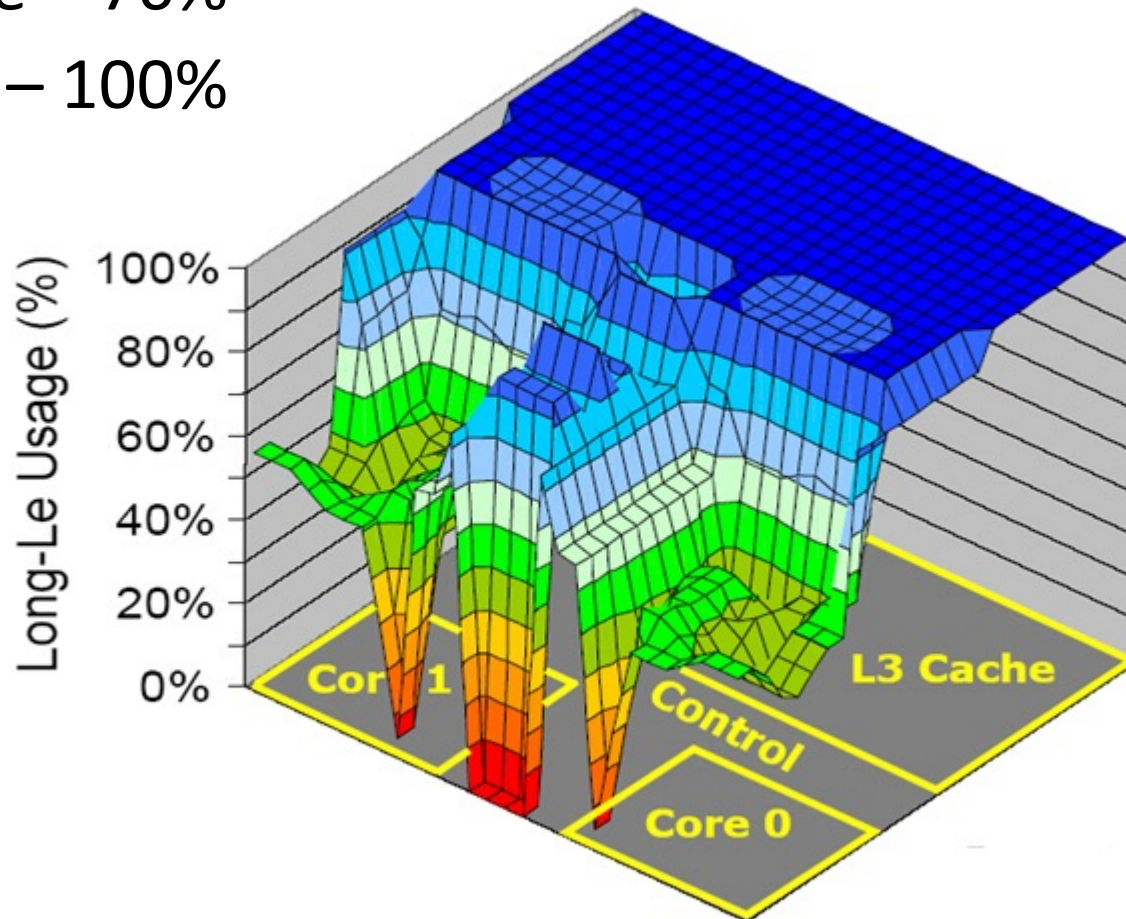
- All transistors can be either nominal or long-Le
- Most library cells are available in both flavors
- Long-Le transistors are about 10% slower, but have 3x lower leakage
- All paths with timing slack use long-Le transistors
- Initial design uses only long channel devices



# Long-Le Transistors Usage

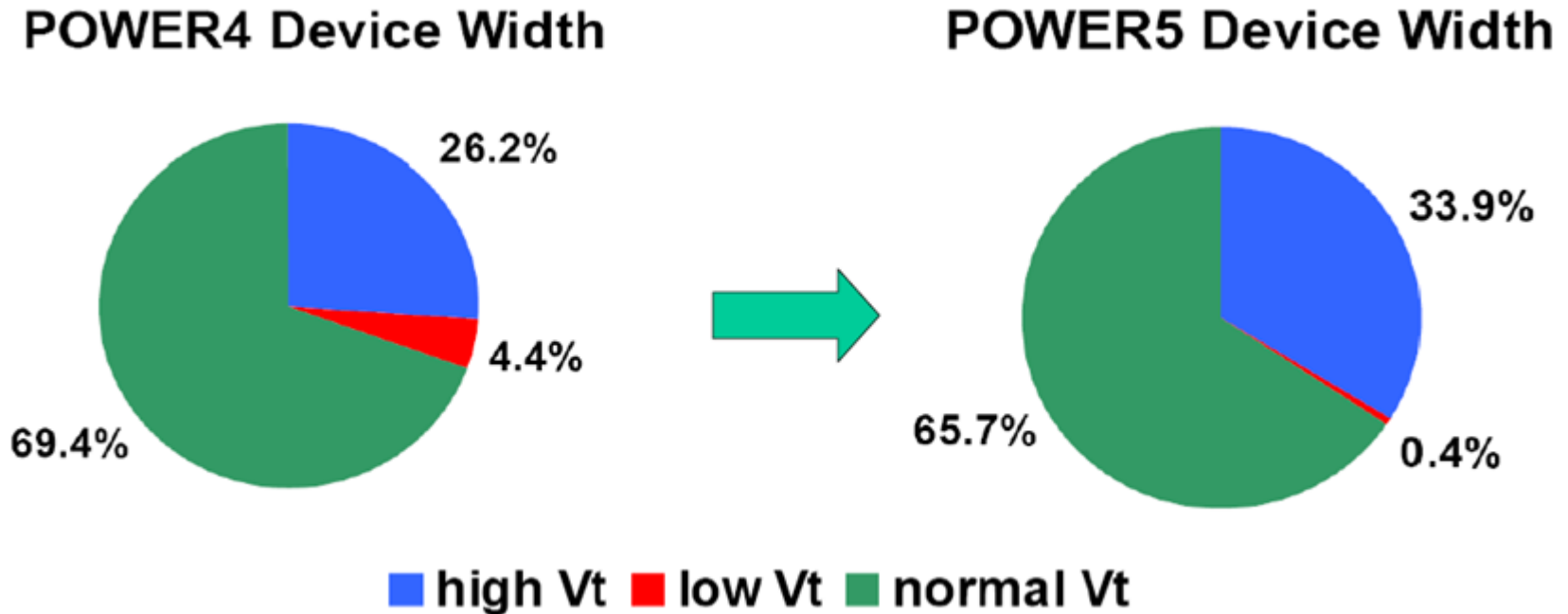
## ■ Long channel devices average usage

- ◆ Cores – 54%
- ◆ Uncore – 76%
- ◆ Cache – 100%



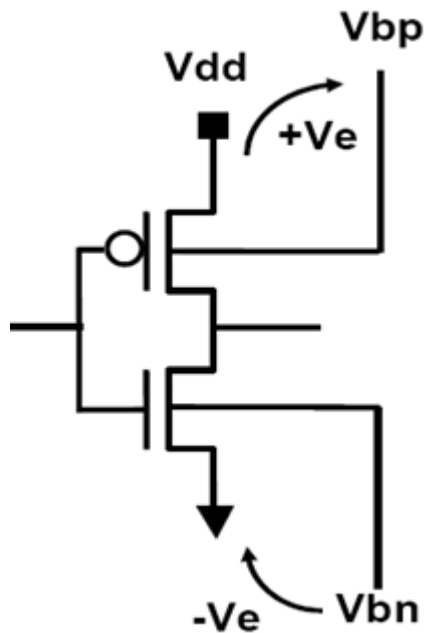
# High- $V_t$ Transistors

- IBM's Power Processors are leveraging triple  $V_t$  process option

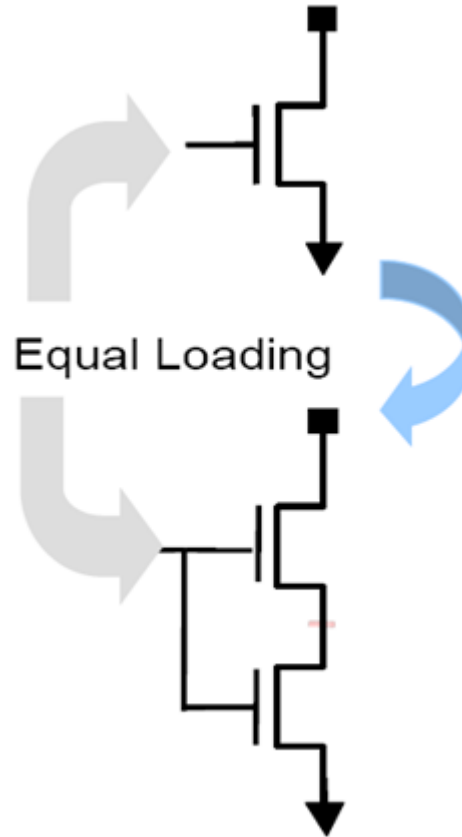


# Leakage Reduction Circuit Techniques

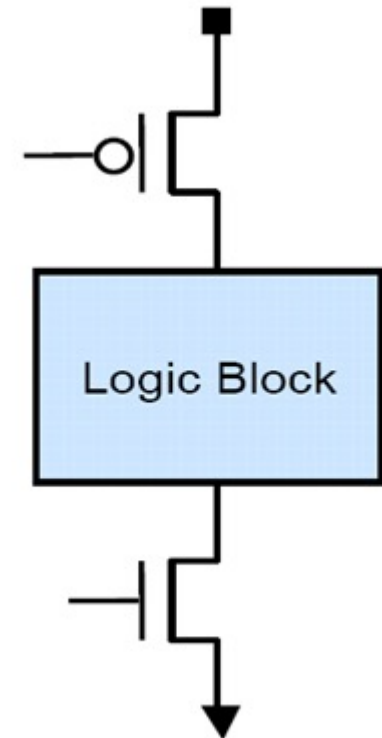
## Body Bias



## Stack Effect

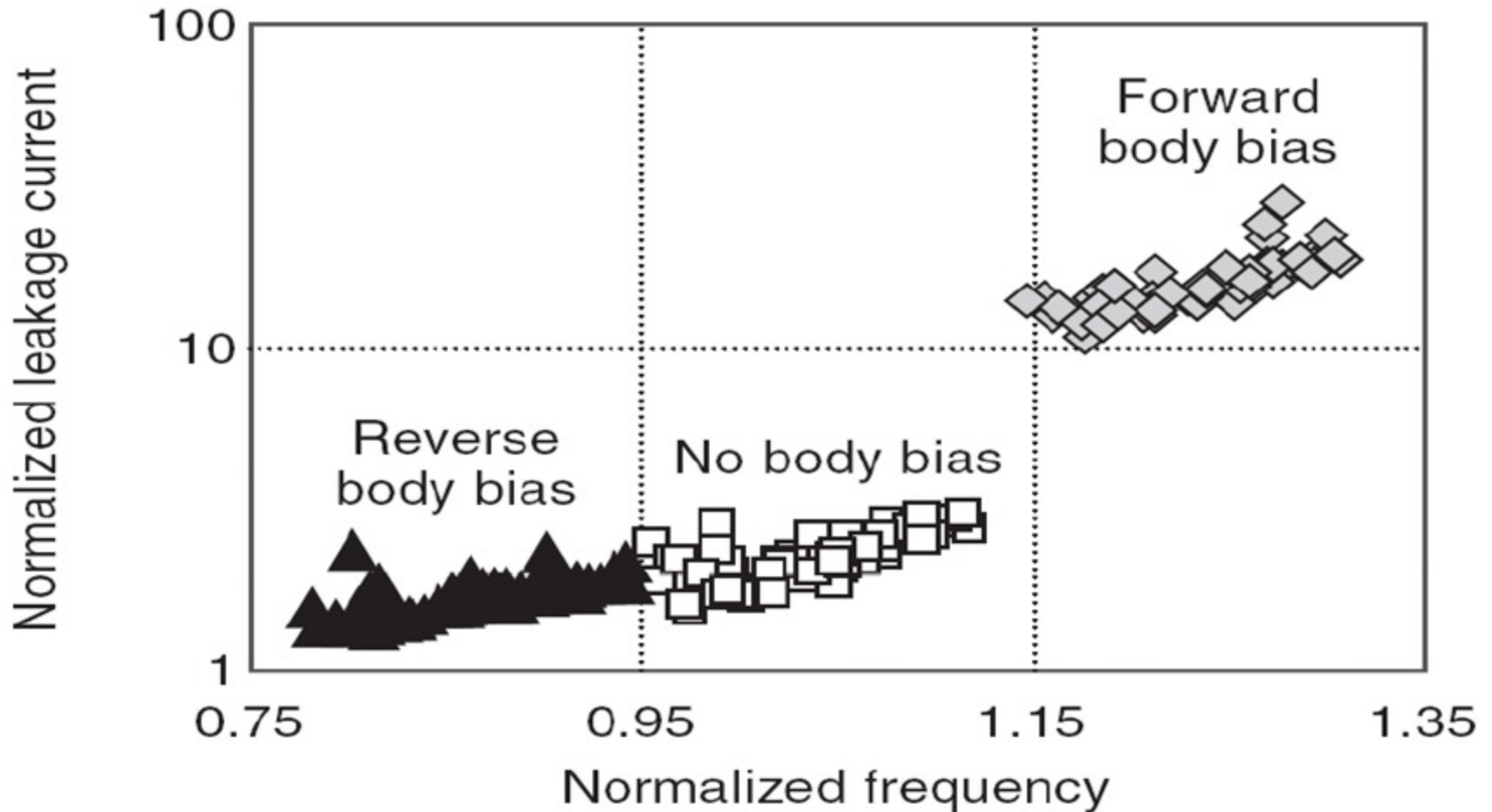


## Sleep Transistor





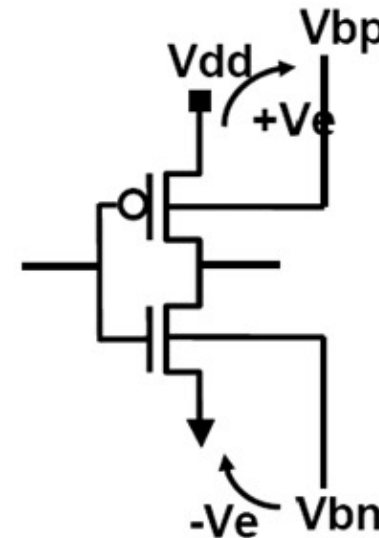
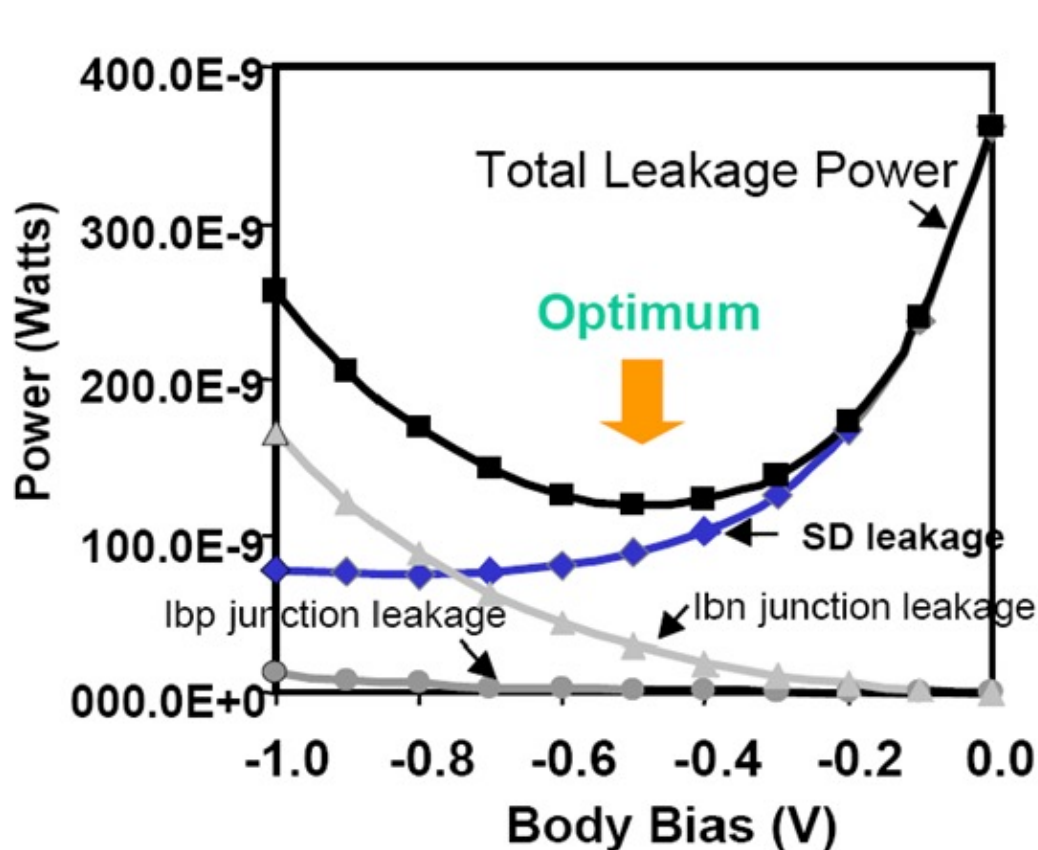
# Body Bias Leakage Reduction





# Scalability of Reverse Body Bias

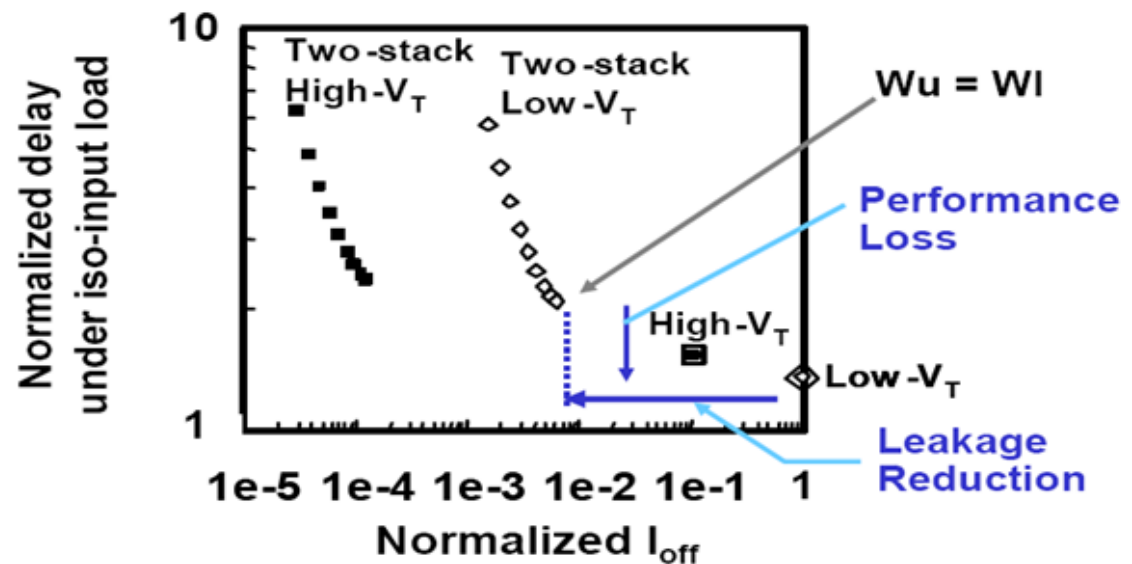
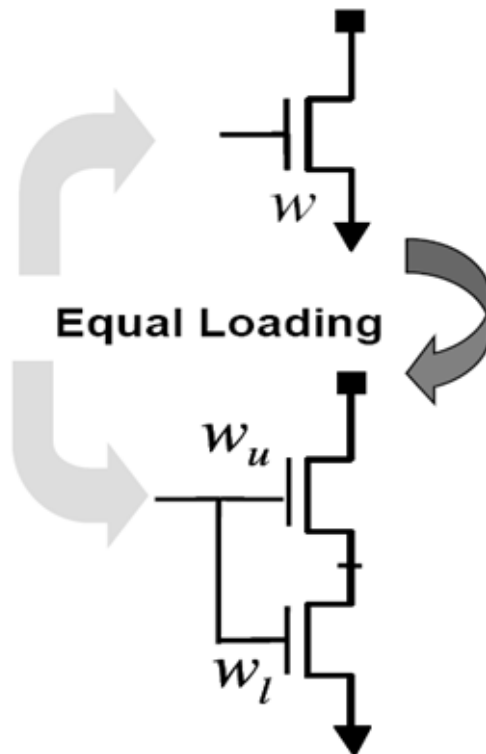
- Reverse body bias is less effective with technology scaling



Tech	0.35 $\mu\text{m}$	0.18 $\mu\text{m}$
Opt. RBB	2V	0.5V
Ioff Red.	1000X	10X



# Stack Forcing

- Force one transistor into a two transistor stack with the same input load
- Can be applied to gates with timing slack
- Tradeoff between transistor leakage and speed



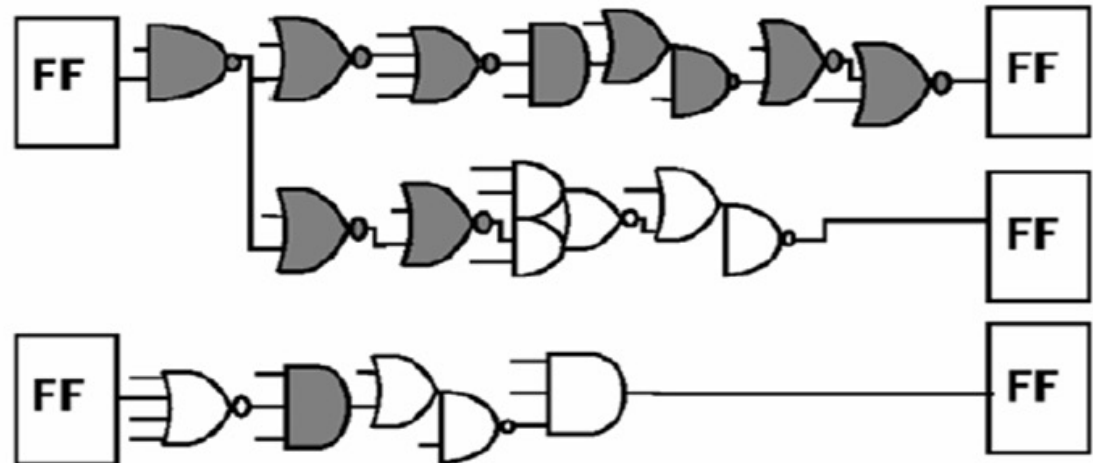
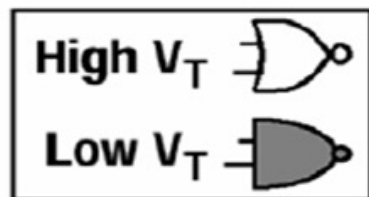
# Stack Effect

## ■ Normalized leakage in stacked MOS

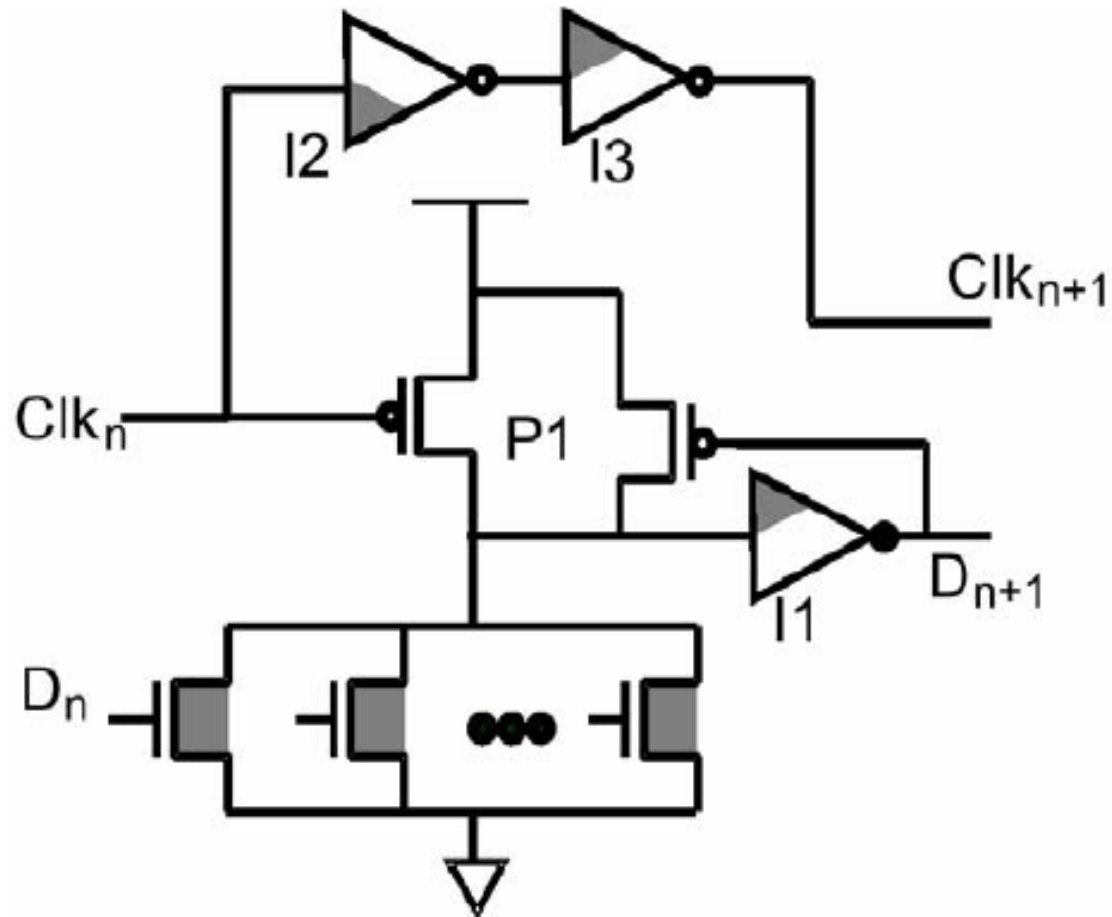
Structure	Without DIBL	With DIBL
 OFF	1	1
 OFF OFF	0.5	~0.1

# Using Multiple Thresholds

- Cell-by-cell  $V_t$  assignment
  - ◆ Not block level
- Allow us to minimize leakage
- Achieve all-low- $V_t$  performance

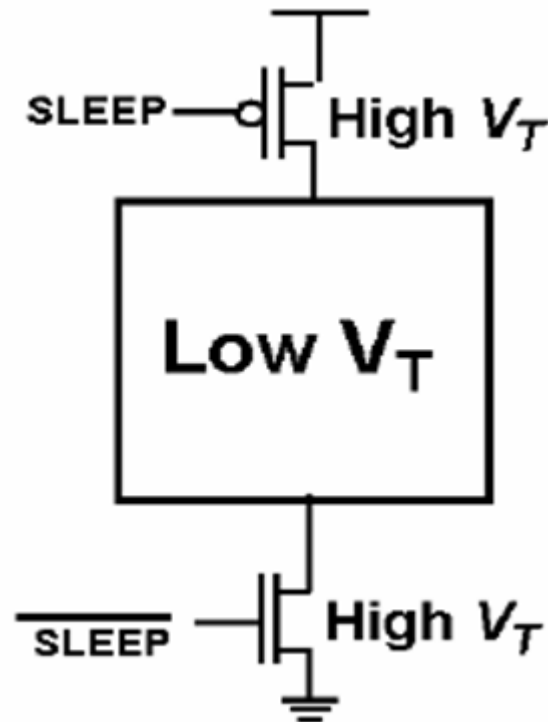


# Dual VT Domino

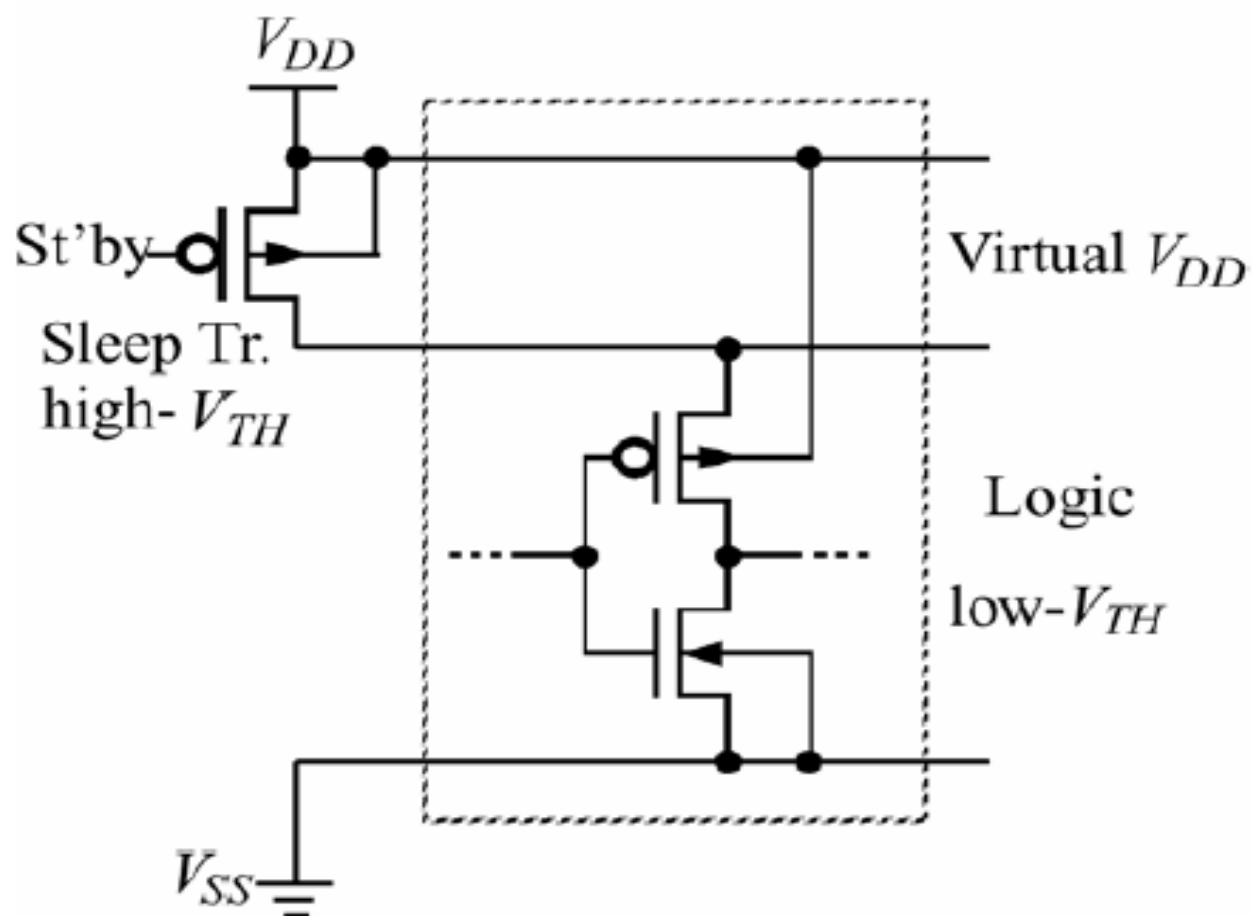


# Techniques for Burst Mode Computation

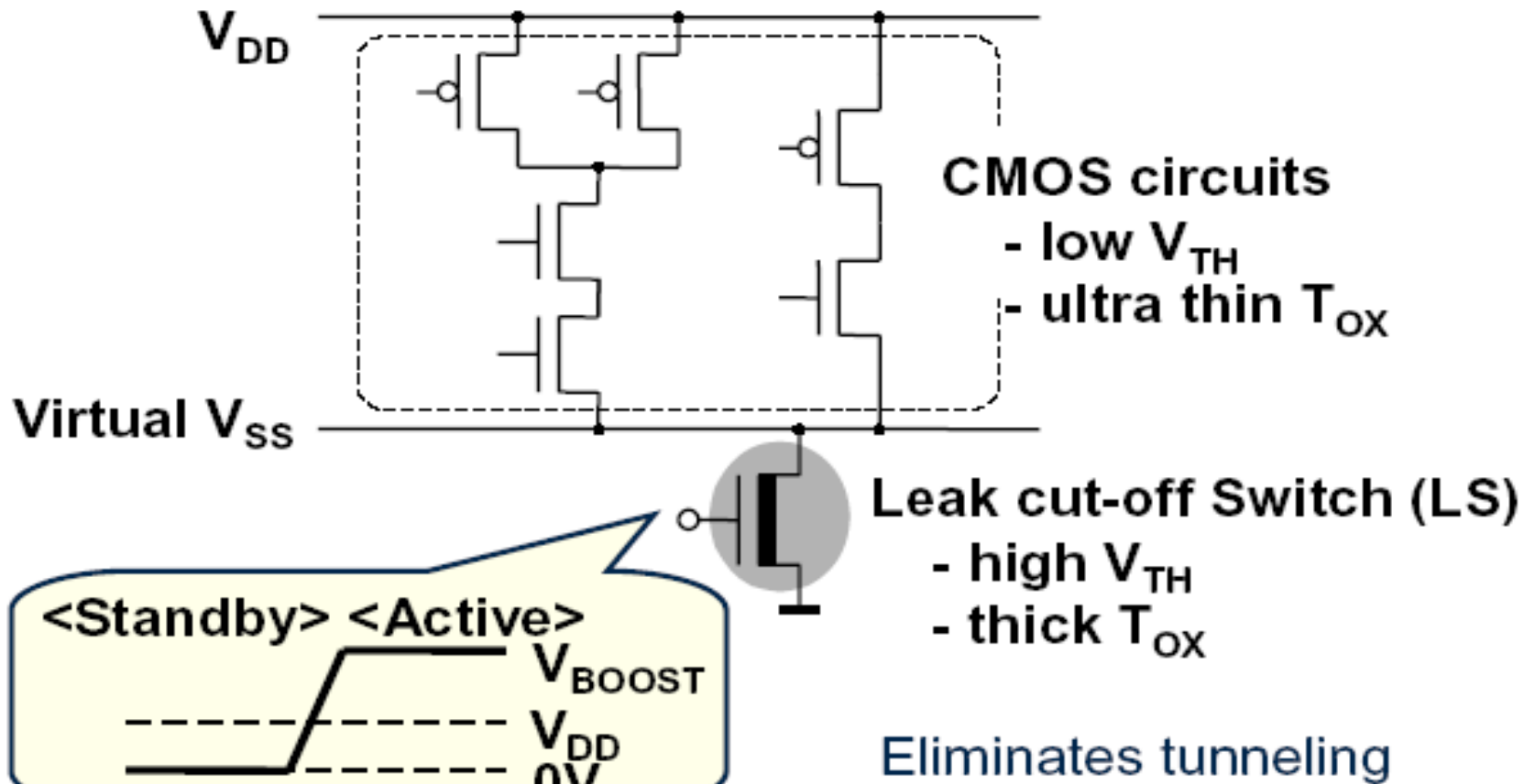
- Multiple  $V_t$  technology
  - ◆ High  $V_t$  transistor sizing issue
  - ◆ Preserving state requires extra transistors



# MTCMOS



# Boosted-Gate MOS (BGMOS)

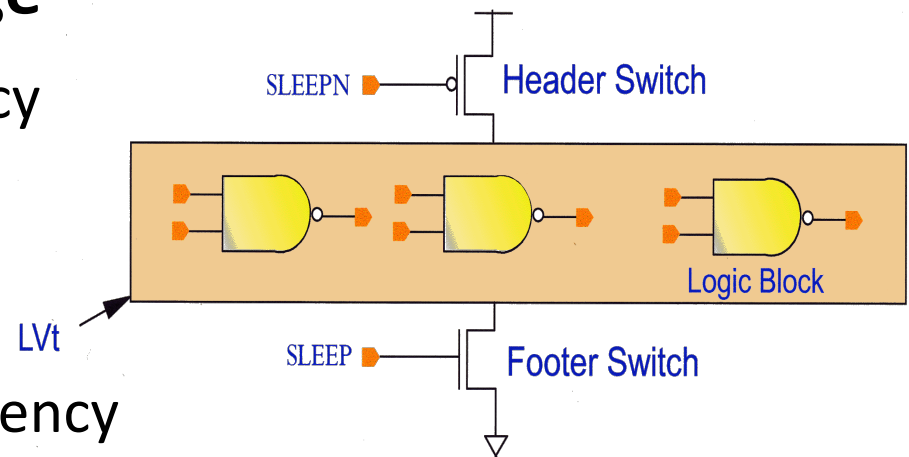


T.Inukai, CICC'00.



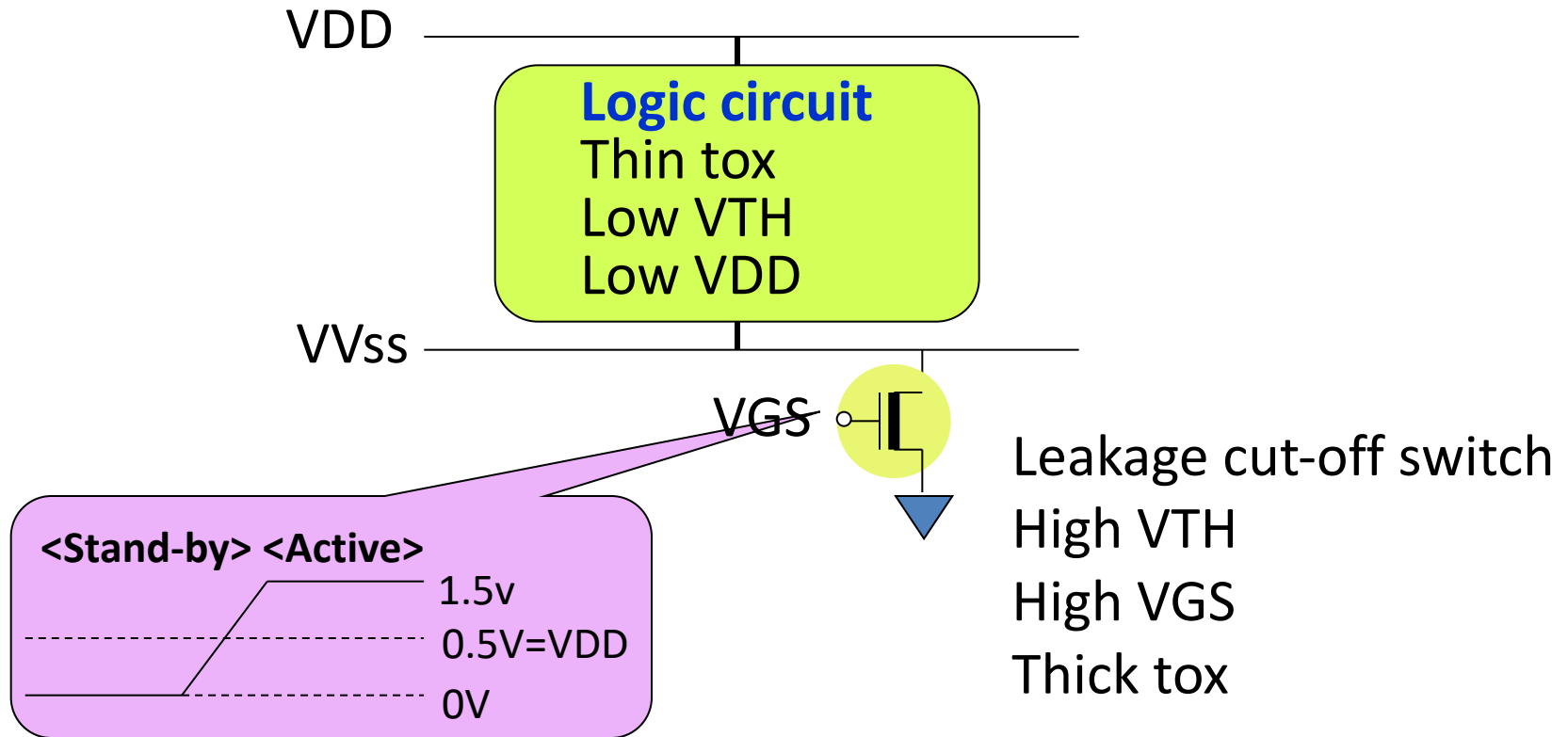
# Standby Mode Leakage Suppression

- **Disconnect inactive logic from supply in standby**
- **Multi-threshold with high  $V_t$  header/footer**
  - ◆ Gate & sub-threshold Leakage current suppression
- **Multi-oxide with thick-oxide header/footer**
  - ◆ Gate leakage current suppression
- **Header/footer gate voltage**
  - ◆ Overdrive – increase frequency
  - ◆ Under-drive – reduce leakage
- **Header/footer well bias**
  - ◆ Forward bias – increase frequency
  - ◆ Reverse bias – reduce leakage



# Stand-by Leakage Reduction

## ■ Through technology-circuit cooperation



Technology provides multiple kinds of MOSFET's and designers make use of the gift.