Power Gating for Ultra-low Leakage: Physics, Design, and Analysis

Ken Choi
Jerry Frenkil
Kimiyoshi Usami
Mark McDermott

Spring 2015

The Leakage Problem: Market Demands

- Increased functionality with longer battery life....

- ... but keep the battery small and lightweight
The Leakage Problem: Advanced Technology

"Leakage will become a major industry crisis, threatening the survival of CMOS itself"

ITRS2005 Executive Summary

Power Review

1. Capacitor or switching power (~50% of total power @ 90 nm)
   - Energy consumed is \( \frac{1}{2} CV^2 \) per transition
2. Short-circuit or internal power (10-15% of dynamic power)
   - When both p and n transistors turn on during signal transition
3. Sub-threshold leakage power (dominates when inactive)
   - Transistors do not turn off completely
4. Gate and diode leakage power (not negligible)
   - Gate oxide tunneling and parasitic source and drain diodes leak to substrate
Leakage Types

- **Standby Leakage**
  - Leakage from blocks in logical standby state
  - Reduction techniques
    - Coarse-grained MTCMOS power gating
    - Body bias control and power supply collapse

- **Active Leakage**
  - Leakage from operationally active blocks
    - Leakage from off-state instances within the active block
    - Gate leakage from active transistors
  - Reduction techniques
    - Fine-grained MTCMOS power gating
    - Multiple threshold voltage cell swapping
    - Long channel devices
    - Input vector control
    - Off-off stacking

Leakage Control Techniques

- **Process oriented techniques**
  - High Vt processes
  - Thick gate oxide processes
  - High-K gate oxide processes

- **Design oriented techniques**
  - Transistor sizing
  - Transistor stacking
  - Input vector control
  - Dual/Multi-Vt cell swapping
  - Body Biasing (VTCMOS)
  - Dynamic voltage scaling (DVS)
  - MTCMOS power gating
Low-Leakage Process Techniques

- **High Vt Processes**
  - Foundry companies often provide low-power technology libraries which have high-Vt standard cells for deep sub-micrometer technologies (below 90nm)
  - Subthreshold leakage is reduced at the expense of reduced drive currents

- **Thick gate oxide processes**
  - Achieve the same Cox with a thicker Tox by using high-k dielectric process

---

**Gate leakage reduction with High-K gate dielectric**

- As gate oxide thicknesses were thinned for 45nm nodes and below, the industry has moved to using high-k dielectrics
  - High-k gate dielectrics enable the process to achieve the same Cox with a thicker Tox

![Gate leakage vs. Oxide Thickness](image)
Low-Leakage Design Techniques Overview

MTCMOS or Guarding

Dual $V_{dd}$ or Multiple $V_{dd}$

VTCMOS, Adaptive Body Bias, Reverse Body Bias, Forward Body Bias.

$V_{dd}$ reduction

Transistor sizing

Stack Effect

Rela4onship between Power and Delay

$Power = P = \frac{1}{2} C_L V_{dd}^2 \cdot \frac{V_{out}}{V_{DD}}$\n
$Delay = \frac{1}{2} \cdot \frac{C_L \cdot V_{DD}}{(V_{DD} - V_{TH})^3}$

- Power is reduced while delay is unchanged if both Vdd and Vth are lowered such as from A to B

4/5/15 EE382M-8 Class Notes
**Rationale of Low Power Approaches**

- More than 60% of gates have their slack larger than 25% of the critical path delay (ISCAS/MCNC/ARM9 functional blocks)
- Exploiting the positive slack

**Trends of Low Leakage Techniques**

- **Temporal granularity (transition time)**
  - Static techniques (infinity) to Dynamic techniques (ns)
- **Spatial granularity**
  - Coarse grained (full-chip level) to Fine grained (block or gate level)
- **Variable granularity**
  - One variable to dual / multiple variables
1. Transistor Sizing

- Shorter width means lower leakage, more delay, and lower dynamic power
- Issues
  - Given delay constraints, finding the optimal size for minimum power
- Pros
  - Leakage reduction in both active and sleep modes
  - Fine-grained optimization is possible
  - Simultaneously optimize width with Vdd and Vth
- Cons
  - Design automation complexity is high
  - Limited amount of leakage reduction
  - Greater variability in drive strength

2. Transistor Stacking

\[ I_{\text{leakage}} \propto 10^{(V_{GS}-V_{TH})/S} \]

- Main factors
  - Negative Vgs, lowered signal rail (Vdd-Vs), lower DIBL (lower Vdd-Vs), and larger Body Effect (negative Vbs)
- Issues
  - Given logic topology, maximizing stacking
- Pros
  - Design complexity is low
  - No impact to technology scaling
  - Area and dynamic power overhead is generally low
- Cons
  - New cell library is needed
3. Input Vector Control (IVC)

- Different input vectors generate different leakage currents, so there must be optimal primary input vectors which lead to minimum leakage power in standby mode

- **Issues**
  - Given logic topology, finding optimal input vector for minimum power

- **Pros**
  - Easy to implement
  - Overhead is low in terms of delay, area, and dynamic power
  - No impact to technology scaling

- **Cons**
  - Relatively less effective on leakage reduction

<table>
<thead>
<tr>
<th>State</th>
<th>( I_{on} ) (( \mu A ))</th>
<th>( I_{leak} ) (( \mu A ))</th>
<th>( I_{off} ) (( \mu A ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0.382</td>
<td>0.306</td>
<td>0.306</td>
</tr>
<tr>
<td>001</td>
<td>0.729</td>
<td>6.359</td>
<td>7.048</td>
</tr>
<tr>
<td>010</td>
<td>0.729</td>
<td>1.375</td>
<td>1.944</td>
</tr>
<tr>
<td>011</td>
<td>5.626</td>
<td>12.677</td>
<td>18.303</td>
</tr>
<tr>
<td>100</td>
<td>0.648</td>
<td>0.006</td>
<td>0.675</td>
</tr>
<tr>
<td>101</td>
<td>5.854</td>
<td>6.559</td>
<td>10.143</td>
</tr>
<tr>
<td>110</td>
<td>5.854</td>
<td>0.006</td>
<td>5.854</td>
</tr>
<tr>
<td>111</td>
<td>28.277</td>
<td>19.015</td>
<td>47.209</td>
</tr>
</tbody>
</table>

4. Dual/Multi-Vth Cell Swapping

- Low-Vth cells on critical paths, High-Vth cells on non-critical paths

- **Issues**
  - Given delay constraints, finding the optimal Vths for minimum power without compromising the delay

- **Pros**
  - No area overhead
  - Leakage reduction in both active and sleep modes

- **Cons**
  - Critical paths are still leaky
  - Limited amount of leakage reduction
  - Weak technology scaling
5. Body Biasing (VTCMOS)

- **Reverse Body Biasing (RBB)**
  - Active Mode: No Bias (= Low Vth) and Sleep Mode: RBB (= High Vth)
- **Forward Body Biasing (FBB)**
  - Active Mode: FBB (= Low Vth) and Sleep Mode: No Bias (= High Vth)

**Issues**
- Given delay constraints, finding the optimal biasing for minimum power

**Pros**
- Useful as a post-silicon tuning method for yield enhancement
- Dynamic approach is suitable for use with DVS

**Cons**
- Less effective at shorter channel length and lower Vth
- Process complexity is high (Triple well is needed)

---

6. Dynamic Voltage Scaling (DVS)

- **Dynamically scale energy/operation with throughput and exploit data dependent computation times to vary the supply voltage**

**Issues**
- Given delay constraints, finding the optimal Vdd and frequency for minimum power

**Pros**
- Very effective to reduce total energy (dynamic and leakage power)

**Cons**
- Difficult to implement
- Additional Control circuits are needed for monitoring and scaling
7. MTCMOS Power Gating

- High Vth transistors gate leakage power during sleep mode for low Vth circuits
- Issues
  - Given delay constraints, finding the optimal clustering and the proper switch size for minimum power
- Pros
  - Most powerful leakage control scheme (10x-100x reduction)
  - Dynamic fine-grained approach can be used to reduce active leakage
- Cons
  - Implementation complexity is high

Leakage Reduction Comparison

1. Transistor Sizing
2. Transistor Stacking / 3. IVC
4. Dual/Multi-Vth Cell Swapping
5. Body Biasing (VTCMOS)
6. DVS (Dynamic Voltage Scaling)
7. MTCMOS Power Gating

- 1.2x – 2x
- 2x – 5x
- 1.5x – 3x
- 5x – 10x
- 3x – 5x
- 10x – 100x
Historical Footprint for Power-Gating

1993

Switched Source-Impedance MOS

1995

MTCMOS

Power-Gating

1999, Mutoh, ACM

2000, Sakurai, BGMOS and Super-Cutoff CMOS

2002, Usami, Selective MTCMOS

2003, Anis, Gate-Clustering

2003, S. Kim, Mode Transition Method

2005, K. Choi/Sakurai, Optimal Zigzag

2007, Frenkil/Venkatraman, CoolPower

Power Gating Variants

- Power Gating current reduction
- Coarse-grained Power Gating
- Fine-grained Power Gating
- Selective Power Gating
- Super-cutoff Power Gating
- Zig-Zag Power Gating
- Advanced Power Gating schemes
Power Gating Current Reduction

- Weak Inversion Current Reduction

\[ I_{\text{LEAK}} \propto 10^{(V_{GS} - V_{TH})/s} \]

Subthreshold swing

\[ V_{GS} - V_{TH} = V_{GS} - (V_{TH0} - \gamma V_{GS} - \delta V_{GS2} - \lambda V_{DS}) \]

Negative \( V_{GS} \)  

High \( V_{TH} \)  

Body bias  

Double gate  

DIBL

Coarse Grained Power Gating

- Power Gating switch cell is a part of the power distribution network
- Power management block controls the power turn on and off of sub-blocks in SoC

Sub-Block Based Power Gating
Fine-Grained Power Gating

- **Switch-in-Cell**
  - Switching transistor is encapsulated as a part of the standard cell

<table>
<thead>
<tr>
<th>Diagram</th>
<th>NAND2(non-footswitch)</th>
<th>fNAND2(footswitch)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYMBOL</td>
<td><img src="source" alt="Diagram" /></td>
<td><img src="source" alt="Diagram" /></td>
</tr>
<tr>
<td>SCHEMATICS</td>
<td><img src="source" alt="Diagram" /></td>
<td><img src="source" alt="Diagram" /></td>
</tr>
<tr>
<td>LAYOUT</td>
<td><img src="source" alt="Diagram" /></td>
<td><img src="source" alt="Diagram" /></td>
</tr>
</tbody>
</table>

[Source: ISSCC. 05, G. Uvieghara]

Selective Power Gating

- **Selective MTCMOS**
  - Low-Vt cells on critical path and high-Vt cells on non-critical paths
  - Apply Power Gating only for the low-Vt cells

[Source: ISSCC. 05, G. Uvieghara]
**Super Cutoff Power Gating**

- Use gate voltage of the PG switch transistor to reduce leakage instead of using high-vt switch transistor
  - Overcome technology scaling issue with high-Vt switch

![Diagram of Super Cutoff Power Gating](image)

[\( V_n \) (negative voltage) generator for NMOS cut-off switch with self-adaptive voltage level detector, charge pump, and oxide-stress-relaxed level shifter, T. Sakurai, 2002]

---

**Zig-Zag Power Gating**

- Use header and footer switches in a ZigZag, alternating on successive instances
  - Overcome long wakeup time for Power Gating scheme

![Diagram of Zig-Zag Power Gating](image)

Conventional cut-off switch several clock cycles to wake-up

Zigzag scheme \( x10-100 \) wake-up speed
**Zigzag Super Cut-off CMOS**

- Combine Zigzag Power Gating and Super Cut-Off Power Gating
  - Overcome long wakeup time and technology scaling issue for general Power Gating scheme

![Zigzag Super Cut-off CMOS Diagram]

<table>
<thead>
<tr>
<th>Node</th>
<th>Wakeup time (1)</th>
<th>Delay of the adder (2)</th>
<th>(1):(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.18μm</td>
<td>0.3ns</td>
<td>1.9ns</td>
<td>16%</td>
</tr>
</tbody>
</table>

J. Tschlan et al., JSSC, pp. 1838-1843, Nov. 2003. 200%

---

**Optimized Zigzag Super Cut-off CMOS**

- Main benefits
  - Optimized IPF (36% of gates on critical path are off-off stacked on avg)
  - No keeper circuit required (zig-zag)
  - Worst-case guarantee (power-gating switch is not shared)
  - Fast wake-up
  - Small ground bounce

![Optimized Zigzag Super Cut-off CMOS Diagram]
Results and Summary

- 15 benchmark: c432, c498, c880, c1355, c1908, c2676, c3540, c5315, c6288, c7552
  - Tech.: 65nm (Vdd = 0.9V, DIFF-VTH: 0.1V, sleep tr. size: 2W; overdrive: 0.1V)
  - % of Low VTH cells / total cells: 23.4% on average (min.: 9.19% / max.: 48.01%)
  - % of IPP over Low VTH cells: 20.1% on average (min.: 11.9% / max.: 30.12%)

<table>
<thead>
<tr>
<th>Power Gating Scheme</th>
<th>Description</th>
<th>Normalized by ORG</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORG</td>
<td>All Low-VTH (Original Circuit)</td>
<td>1.0</td>
</tr>
<tr>
<td>HVT</td>
<td>All High-VTH (Diff: 0.1 volt)</td>
<td>0.06</td>
</tr>
<tr>
<td>DUAL</td>
<td>Dual VTH (% of Low VTH: 23.4%)</td>
<td>0.32</td>
</tr>
<tr>
<td>MT</td>
<td>Gate-Level MTCMOS (2W-sleep tr. sizing)</td>
<td>0.07</td>
</tr>
<tr>
<td>IPF</td>
<td>Input Phase Forcing Only (Off-off Stacking only)</td>
<td>0.37</td>
</tr>
<tr>
<td>SMT</td>
<td>Selective MTCMOS (DUAL + MT)</td>
<td>0.06</td>
</tr>
<tr>
<td>OZ (No overdrive)</td>
<td>Proposed Optimal Zigzag (DUAL + IPF + ZZ)</td>
<td>0.08</td>
</tr>
<tr>
<td>OZ (Overdrive)</td>
<td>Proposed Optimal Zigzag (Overdrive Voltage: 0.1V)</td>
<td>0.02</td>
</tr>
</tbody>
</table>

70% Decrease in Leakage 9.4% Decrease in Delay 37% Decrease in Area

Advanced Fine-Grained Power Gating

- Use Clock Gating enable signal to control Power Gating circuits
  - Requires fast wakeup and small speed loss

[Diagram of Advanced Fine-Grained Power Gating]
Advanced Power Gating Scheme with Nanotubes

(a) Low-Power Design for CMOS (MTCMOS Power Gating, 3 Stage Inverter, 10x leakage power reduction in 65 nm)

(b) Hybrid PowerGating Design with CMOS and CNT (Carbon Nano Tubes)

(c) 3 Stage CNT FET Inverter (100x leakage reduction and 10x fast speed over 65 nm CMOS design)

One Last Thing . . .

- PVT (Process, Voltage, and Temperature) variations should be very carefully considered with the leakage reduction techniques

![PVT Variations](image)
Quick Summary: Leakage Physics and Control

- Leakage has become a very tough problem
- Leakage worsens with each new process generation
- Designers must solve the leakage problem
  - Process engineers do not have realistic solutions on the horizon
- Multiple design solutions exist for reducing leakage
- MTCMOS power gating will become the dominant solution for ultra-low Leakage.

Power Gating Design Issues

- Logic Design
- Physical Design
- Design Verification
- Tools, Flows, and Methodologies
Power Gating Issues – Logic Design

- **High-side or Low-side switching**
  - Use of header switches to gate the power supply or footer switches to gate ground

- **Domain partitioning**
  - Partitioning the design into sections with always-on power and those with gated power

- **Signal interfacing**
  - Conditioning the outputs of power-gated blocks such that they can not float

- **State retention**
  - Retaining some or all of the internal state when power is removed

---

High-side or Low-side Switching

![Diagram of high-side or low-side switching](image-url)
High-side or Low-side Switching

■ Header: PMOS device used to gate power supply
  — Advantages
    • Less susceptible to gate leakage than footers
    • Can be used with multiple supply voltages
  — Disadvantages
    • ~2X larger than equivalent resistance footer

■ Footer: NMOS device used to gate the ground
  — Advantages
    • ~1/2 the size of equivalent resistance header
  — Disadvantages
    • Does not reduce gate leakage

Refresh with contemporary CMOS technology.
1. the 2X area is no longer applicable; PFET IDSATs are now comparable to NFETs.
2. The virtual node will get closer to GND with a PFET pwr gate vs. an NFET pwr gate?
3. Switching performance is better? No pull-down de-biasing with solid GND grid.
4. PFET switches allows the use of different VDD values to different blocks in the same SoC design.

Domain Partitioning

■ Which modules must be always powered on?
  — Assign these modules to the always-on domain

■ Which blocks can be power-gated?
  — Assign these to the power-gated domain
    • Switch enables must be available
  — How much leakage will power gating save? Is it worth the effort?
    • How much does the module leak when powered?
    • How long is the module idle time?

■ How many separate modules are worth power-gating?
  — Only a few large domains: Coarse-grained power gating
  — A large number of very small domains: Fine-grained power-gating
Domain Partitioning: Coarse / Fine Grained

- **Coarse Grained**
  - Typically used to reduce *standby* leakage power

- **Fine Grained**
  - Useful for reducing *active* leakage
    - Small logic modules can be powered off, even while rest of module is active
  - Finding / creating enables can be difficult
  - Physical design may be difficult
  - Number of isolation cells needed can be substantial

Signal Interfacing

- Power-gated logic outputs become high-impedance drivers when switches are opened....

- ...but high-impedance drivers must not drive powered logic
  - High impedance nets can float to intermediate voltages resulting in large through currents in receiving logic
Signal Interfacing – Different Types

- **Buffers: AND, OR gates**
  - AND used with footers; pulls signal low
  - OR used with headers; pulls signal high

- **Pull-up, Pull-down**
  - Pull-up used with footers; pulls net high
  - Pull-down used with headers: pulls net low

- **Half-latch**
  - Maintains most-recent state of output

Signal Interfacing for Multiple Domains

- **All outputs of power-gated blocks must be isolated**
State Retention

- Remember critical state so that when power is turned back on the logic resumes operation from the point at which power was removed

- Key question:
  - How much / which state needs to be retained?

- Answer:
  - It depends on the application
    - Generally, data does not need to be saved
    - But control registers must be saved

State Retention Methods

- Scan-out(scan-in)
  - State in power-gated domain is scanned out into an always-on domain prior to opening switches
  - State is scanned back in after switches are re-closed
  - Power manager design can be complex
  - Lengthy power-down and power-up time

- State retention registers (SRFF)
  - Requires both switched and always-on power
  - Fast power-down and power-up

- State retention voltage
  - Reduce supply to ~ zero, but not so low as to lose state
  - Medium speed power-down and power-up
State Retention Register Design

- **Balloon Retention Register**
  - State is saving in an always-on shadow register
  - Main register uses low-Vt devices
  - Balloon uses high-Vt devices

- **Always-on Slave**
  - Modified master-slave flip-flop
  - Slave is always powered
  - Master latch, clock buffers, and output buffers are power-gated

State Retention Tradeoffs

<table>
<thead>
<tr>
<th>Method / Parameter</th>
<th>Area</th>
<th>Power-down time</th>
<th>Recovery Time</th>
<th>Leakage Reduction</th>
<th>Power manager design</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scan-in / Scan-out</td>
<td>Additional logic needed for scanning state in/out</td>
<td>Slowest – must scan state out</td>
<td>Slowest – must scan state back in</td>
<td>Some leakage from additional logic</td>
<td>Most complex</td>
</tr>
<tr>
<td>SRFF</td>
<td>Biggest registers, but little additional logic</td>
<td>Fastest</td>
<td>Fastest</td>
<td>Registers still leak, amount depends on register design</td>
<td>Least complex</td>
</tr>
<tr>
<td>Reduced voltage</td>
<td>Least</td>
<td>Almost as fast as SRFF</td>
<td>Moderate – depends upon voltage regulator</td>
<td>Depends upon voltage for virtual rail</td>
<td>Depends upon number of voltage levels</td>
</tr>
</tbody>
</table>
Power Gating Issues – Physical Design

- Switch placement
- Switch sizing
- Rush current & Wakeup time control
- Library requirements

Switch Placement Options

- **Switch-in-cell**
  - Each standard cell has a switch transistor embedded in the cell itself -> no standalone or separate switch cells
  - Each instance is power-gated

- **Grid of switches**
  - Switches are arrayed across the power-gated block

- **Filler switches**
  - Switches placed in available slots, otherwise occupied by filler cells

- **Ring of switches**
  - Ring separates *internal virtual rails from external real rails*
Switch Placement Tradeoffs

<table>
<thead>
<tr>
<th>Placement Type</th>
<th>Merits</th>
<th>Drawbacks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch-in-cell</td>
<td>Easy P&amp;R; No floorplanning required</td>
<td>Very large area overhead</td>
</tr>
<tr>
<td>Grid of switches</td>
<td>Area efficient</td>
<td>Must route a third rail</td>
</tr>
<tr>
<td>Filler switches</td>
<td>Easy P&amp;R; No floorplanning required; Area efficient</td>
<td>Specialized library layouts required; IR droop issues due to irregular switched power grid</td>
</tr>
<tr>
<td>Ring of switches</td>
<td>Useful for hard layout IP</td>
<td>Significant area overhead</td>
</tr>
</tbody>
</table>

Switch Sizing

- **Switches must be sized to meet electrical specs**
  - Voltage drop spec
  - Timing impact

- **Conceptual model – switch as a resistor in the power delivery network**

- **Critical issue: current estimation**

![Diagram of switch placement tradeoffs and switch sizing](image-url)
### Switch Sizing Methods

<table>
<thead>
<tr>
<th>Technique</th>
<th>Current Calculation Method</th>
<th>Merits</th>
<th>Drawbacks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Current Method (ACM)</td>
<td>Average (DC) currents assumed</td>
<td>Simple – avoids dynamic current analysis</td>
<td>Unrealistic; not worst case</td>
</tr>
<tr>
<td>Switch in Cell</td>
<td>Discharge currents simulated during cell characterization</td>
<td>Easy to characterize</td>
<td>Large area overhead; significant library development</td>
</tr>
<tr>
<td>Mutually Exclusive Switching</td>
<td>Discharge currents obtained from input vector simulation</td>
<td>Provides upper bound on switch sizing</td>
<td>Potential oversizing</td>
</tr>
<tr>
<td>Min-Max Window</td>
<td>Trapezoidal discharge waveform over mix-max switching window</td>
<td>Clustering minimizes max simultaneous switching</td>
<td>Potential oversizing; not worst case</td>
</tr>
<tr>
<td>Event Driven (VEDA)</td>
<td>Composite discharge waveform composed of currents from all</td>
<td>All possible events are considered; sizing</td>
<td>Potential oversizing</td>
</tr>
<tr>
<td></td>
<td>switching events</td>
<td>covers worst case conditions</td>
<td></td>
</tr>
</tbody>
</table>

### Switch Sizing Tradeoffs

**Fundamental tradeoff: area, performance, leakage**
- Voltage drop across switch impacts performance and leakage

<table>
<thead>
<tr>
<th>Switch Size</th>
<th>Voltage Drop</th>
<th>Performance Impact</th>
<th>Area</th>
<th>Off-state Leakage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bigger switches</td>
<td>Smaller</td>
<td>Smaller</td>
<td>Larger</td>
<td>Larger</td>
</tr>
<tr>
<td>Smaller switches</td>
<td>Larger</td>
<td>Larger</td>
<td>Smaller</td>
<td>Smaller</td>
</tr>
</tbody>
</table>

**Implications**
- For least leakage – choose smallest switches without timing violations
- For smallest area – choose smallest switches without timing violations
- For fastest chip – choose biggest switches that area will allow
Switch Sizing Tradeoff Summary

- Larger switches
  -> More area, more leakage, higher performance
- Smaller switches
  -> Less area, less leakage, lower performance

Rush Current and Wake-up Time

- Rush Current: the current that flows through the switches when the switches transition from open to closed
- Wakeup Time: the time required for the virtual rail to settle down to an operational voltage after the switches have been closed

Control Motivation

- Minimize peak current flow when switches are enabled
  - No EM problems
- Minimize wake-up time
  - Switched logic must be operational within specified recovery time
- Minimize voltage drop in always-on logic
Rush Current & Wakeup Time Tradeoff

- Closing switches rapidly => large rush current
- Closing switches slowly => lengthy wakeup time

Rush Current

<table>
<thead>
<tr>
<th>Rush Current</th>
<th>t</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rapid switch closure</td>
<td></td>
</tr>
<tr>
<td>Slow switch closure</td>
<td></td>
</tr>
</tbody>
</table>

Rush Current Control Techniques

- **Single pass cascade**
  - Sleep enable traverses switch network in one direction

- **Two pass cascade**
  - Sleep enable traverses switch network in two directions – first with small switches and second with large switches

- **Staggered turn-on**
  - Multiple sleep enables turn on with staggered delays according to sleep controller outputs
Power Gating Library Requirements

- **Power Gating compatible libraries must contain**
  - **Switch cells**
    - Different sized switches desirable
    - Switches with and without built-in delay buffers
  - **State retention registers**
    - May / may not be necessary depending upon retention strategy
  - **Isolation cells**

- **Special Case: Switch-in-cell implementation**
  - A power-gated version of each logic cell is needed
  - No switch cells are needed
    - Since a switch is embedded in each logic cell
  - Isolation cells may / may not be needed
    - Depends upon whether isolation is built-in to power-gated logic cells

Power Gating Issues - Design Verification

- Logical verification
- Voltage drop verification
- Rush current and wakeup time verification
Logical Design Verification

- Are the switches connected and controlled correctly?
  - Are always-on blocks in fact always powered?
  - Do the power-gated blocks turn-off when and as expected?
  - Do the power-gated blocks turn-on when and as expected?
    - Each power state transition should be simulated / verified

- Are all signal outputs of power-gated blocks isolated?

- Can state be stored and restored?
  - Are retention controls controllable and observable for test?
  - Is there a minimum power-down time before power gating controller can attempt to power up?

Voltage Drop Verification

- Verify voltage drop during active mode operation
  - Check dynamic voltage drop
    - Virtual rail dynamic voltages must be within voltage drop limit
    - Real rail dynamic voltages must be within voltage drop limit
  - Check current densities
    - Virtual rail current densities must be below EM limit
    - Real rail current densities must be below EM limit
    - Switch currents must be below EM limit

- Verify voltage drop during mode transitions
  - Check rush currents and wake-up times
Rush Current & Wakeup Time Verification

- Verify rush currents for powering-up operation
  - Analyze transient current flow when switches close
  - Must be performed concurrently with dynamic voltage drop analysis
    - Voltage drop affects current flow, and current flow affects voltage drop
  - Check currents
    - Peak currents must be below current limit
  - Check voltages
    - Max voltage drop on real rail must be below voltage drop limit

- Verify wakeup times
  - Check time required for power-gated logic to see a steady voltage
  - Check time required for always-on logic to see a steady voltage

Power Gating Design Flow and Design Tools

- Power Gating design flow
- Power Gating commands for power intent
- Power Gating tool requirements
### Generalized Low Power Design Flow

- **System Level Design**
  - Goal: Explore architectures, algorithms, and protocols
  - System Level Design

- **RTL design**
  - Goal: Generate RTL to match system level design
  - RTL Design

- **Implementation**
  - Goal: Synthesize RTL to gates, floorplan, place & route
  - Implementation

- **Verification**
  - Goal: Verify correct functional and electrical operation
  - Verification

### Power Gating Aware System Level Design

- Define power states and domains
  - Number of different domains
- Define power-down and power-up protocols
- Define interface and isolation protocols
- Define retention strategy
  - Save and restore methods
  - Full or partial state retention
- Decide upon high-side or low-side switching
Power Gating Aware RTL Design

- Specify which registers must maintain state
- Design power gating controller
  - Generate sleep enable signals
- Design rush current control scheme
- Simulate RTL to confirm power gating behavior
  - Always on, power-up, & power-down
- Analyze power to determine if design will meet power specs
- Create power gating design constraints in CPF / UPF

Power Gating Aware Implementation

- Synthesize RTL to gates
  - Specify which registers must maintain state
- Floorplan, P&R design
- Insert, size, & place switches
- Buffer and distribute sleep enables
- Tune rush current control circuit
### Power Gating Aware Verification

- **Power specs**
  - Leakage power
  - Dynamic power
- **Voltage Drop**
  - Real rails & Virtual rails
- **EM**
  - Real rails & Virtual rails
- **Rush Currents**
- **Wakeup time**
- **Timing effects**

### Power Gating Design Constraints

- **Specify power intent to multiple tools with a single constraint language**
  - Example: Power domains, logical and physical
- **Two different (but very similar) TCL-based languages**
  - CPF: Common Power Format
  - UPF: Unified Power Format
- **Some CPF commands**
  - set_*_commands: general commands
  - define_*_cell commands: library cell descriptions
  - create_*_rule commands: design intent
  - update_*_rule commands: implementation directives
Power Intent

- **Power domains**
  - Logical partitions
  - Physical partitions

- **Power logic**
  - Switches
  - Isolation cells
  - State retention logic

- **Power modes**
  - Mode definitions
  - Transition definitions

---

CPF Example Snippet

```c
#define power domains
create_power_domain -name PDcore -default
create_power_domain -name PDau -instances alu_inst/aui \
  -shutoff_condition pcu_inst/pau[2]

#define condition which triggers state retention
create_state_retention_rule -name sr_rule -domain PDrf \
  -restore_edge {!pcu_inst/[rf[1]]}

#define which signals will be isolated, under which condition, and how
create_isolation_rule -name iso-rule1 -from PDau \ 
  -isolation_condition {!pcu_inst/pau[0]} \ 
  -isolation_output low

#define voltages
create_nominal_condition -name high -voltage 1.2
create_nominal_condition -name low -voltage 0.0

#define power modes
create_power_mode PM1 -default -domain_conditions {PDcore@high PDau@high}
create_power_mode PM2 -domain_conditions {PDcore@high PDau@low}
```

---

System Level Design
RTL Design
Implementation
Verification
Power Gating SoC Tool Requirements

- **Key SoC tools must be Power Gating aware**
  - RTL functional simulator
    - Needed to simulate power gating power-up / power-down behavior
    - Must be CPF or UPF compliant to understand power intent
  - RTL power analyzer
    - Needed to analyze module level power consumption and to predict power savings from Power Gating and other low power techniques
  - Synthesizer
    - Needed to insert isolation cells and SRFFs where appropriate
  - Switch sizer
    - Power Gating specific tool needed to correctly size switches
  - Placer and router
    - Needed placing and connecting switches according to desired topology
  - Voltage drop analyzer
    - Needed to analyze voltage drop across switches and verify rush current and wake-up time behavior.

- **Power Gating awareness can be controlled by CPF / UPF**

Quick Summary: Power Gating Design Issues

- **Power Gating presents unique issues to designers**
  - Logic design, physical design and verification

- **Unfamiliar trade-offs must be confronted efficiently**
  - Headers vs footers
  - Switch size vs area vs leakage reduction vs delay
  - Switch placement vs area vs ease of layout
  - Power controller complexity vs power verification

- **New tools and methodologies are required**
  - Existing tools are evolving to become power aware
  - But new methodologies accompany the new tools
  - Usage of CPF / UPF will become requirements for efficient flows
References (1 of 3)

4. L. Wei, Z. Chen, K. Roy, Y. Ye, and V. De, “Mixed Vth CMOS circuit design methodology for low power applications,” DAC, 1999
References (2 of 3)


References (3 of 3)