Leakage minimization design techniques
Circuit Techniques

CIRCUIT TECHNIQUES TO REDUCE LEAKAGE POWER

High performance circuits usually consist of large gates and parallel architectures with large amounts of logic working in parallel

This is the worst case for leakage power

Many modules have bursty computation: it is of interest to conceive methods that can reduce leakage power consumed by those circuits

Circuit techniques reduce leakage utilising idleness or slack and without impacting performance
Circuit techniques

CATEGORIZED BASED ON WHEN AND HOW THEY REDUCE LEAKAGE

**DESIGN TIME TECHNIQUES**

Exploit the delay slack in non-critical paths to reduce leakage.
Static techniques. Main technique: DUAL Vth static assignment

**RUN TIME TECHNIQUES**

**STANDBY LEAKAGE REDUCTION**
Entire subcircuits put to low leakage mode when computation is not needed
- Natural stacking
- Sleep transistors
- Adaptive Body Bias

**ACTIVE LEAKAGE REDUCTION**
Slow down the system by dynamically changing the Vth to reduce leakage when max. performance is not needed
Gate-level Dual-Vth Design

- Gates on the critical path are inferred with low-Vth library gates
- Gates on non-critical are inferred with high-Vth library gates

*High performance and low leakage simultaneously achieved at some fabrication cost*
Gate-level Dual-Vth Design

This causes a balance between timing paths that may give rise to critical path outliers due to process variations.
Gate-level Dual-Vth Design

The critical path is preserved while compacting the distribution of timing paths

Useful to reduce leakage during standby and active modes of operation

CHALLENGE: how do such multiple critical paths behave in the presence of process variations?
Case study

4-way VLIW processor core (VEX architecture)

- 4 Pipeline Stages
  - Instruction Fetch
  - Instruction Decode
  - Instruction Execute
  - Write Back

- Register File and Execute Bypass Network

- 4 slots in EX stage, each comprising
  - ALU
  - Shifter
  - Compare Unit
  - Multiplier

- Design fully synthesized (including RF) on STMicroelectronics 65nm low-power technology

| Max Freq.: 256 MHz | Area: 0.31 um2 | Power: 30.8mW |

Max Freq.: 256 MHz | Area: 0.31 um2 | Power: 30.8mW
About 10% random variations, +5% Worst Case systematic variations applied

Critical path distribution for each pipeline stage

- All Pipeline stages violate the timing constraint
- The global critical path is almost always in the EX stage
- Clock frequency degraded up to 10% (μ + 3σ of EX distribution)
Case study

- The EX stage suffers from the largest deviation of the mean of the distribution from the nominal critical path

High number of signal paths in a short time window close to critical path. The critical path in this case is likely to be determined by an outlier.

This is a best case since dual-Vth not applied!

- The EX stage exhibits the lowest ratio of variance-to-mean

Path delays are determined by taking an aggregate sum of each gate’s delays in each path.
Dual-Vth CMOS: the physical viewpoint

- How to manufacture High Vth TNs?
  - changing (increasing) channel doping density

- 2 additional masks required, resulting in higher process cost

- due to non-uniform distribution of the doping density, difficult to achieve dual-threshold devices when threshold voltages are close to each other (which is the case for nanoscale devices)

- High-Vth could be achieved with strong Halo (e.g., moving the position of the lateral peak closer to the centre of the channel)… while largely increasing IBTBT
Dual-Vth CMOS: the physical viewpoint

- How to manufacture High Vth TNs?
  - Higher oxide (Tox) thickness
    - reduces subthreshold leakage as well as gate oxide tunnelling (this latter exponentially)
    - reduces also gate capacitance, thus proving beneficial also for dynamic power
  - CHALLENGE: in nanoscale devices, higher Tox worsens SCEs, thus even increasing the subthreshold leakage!! As a workaround, high-Tox devices should come with longer L
  - Advanced process technology is required for fabricating multiple Tox CMOS
Dual-Vth CMOS: the physical viewpoint

How to manufacture High Vth TNs?

- higher transistor channel length
  - for short channel TNs, the Vth increases with increasing channel length (Vth roll-off). The effect can be directly exploited to get multiple Vths by means of differentiated L
  - conventional CMOS process required for multi-L design

CHALLENGE:

- The Vth roll-off is quite sharp
- It is non-trivial to control Vth close to the minimum feature size

CHALLENGE:

- Increased L augments gate capacitance, hence power (beyond reducing speed)
Wrap-up on dual-Vth design

With the increase in Vth variation and supply voltage scaling it is becoming difficult to maintain sufficient gap among low-Vth, high-Vth and the supply voltage required for dual-Vth design. This adds up to the process variation challenge to contain Vth variation.
Runtime techniques

Key idea

*Put circuits in in a low-leakage standby mode when they are not used*
Transistor stacking

Increasing the source voltage of an NMOS transistor reduces sub-threshold leakage current exponentially due to

- Negative $V_{gs}$
- lower $V_{ds}$
- body effect
- reduced DIBL

$$I_{sub} = \frac{W}{L} \mu e V^2 \nu T_{sth} \left( \frac{V_{GS} - V_{th} + \eta V_{DS}}{n V_T} \right) \left( 1 - e^{ \frac{-V_{DS}}{n V_T} } \right)$$

Transistor stacks provide a natural exploitation of this effect
Transistor stacking

Stacked and turned off transistors reduce subthreshold leakage current substantially since top stack transistors end up having negative $V_{gs}$ (self-reverse transistor biasing)

Overall, by switching off more transistors in a stack, the subthreshold leakage current decreases. However, diminishing returns are observed!
Transistor stacking

The voltages at the internal nodes depend on the input applied to the stack.

Maximizing the number of off transistors in a stack by applying proper input vectors can reduce standby leakage of gates (NOR,NAND) and functional block.

**Input vector control for gates with natural stacks**

<table>
<thead>
<tr>
<th>Circuit</th>
<th>Input vector</th>
<th>Iddq(nA)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 input NAND</td>
<td>ABCD = 0000</td>
<td>0.60</td>
<td>Best</td>
</tr>
<tr>
<td></td>
<td>ABCD=1111</td>
<td>24.1</td>
<td>Worst</td>
</tr>
<tr>
<td>3 input NOR</td>
<td>ABC=111</td>
<td>0.13</td>
<td>Best</td>
</tr>
<tr>
<td></td>
<td>ABC=000</td>
<td>29.5</td>
<td>Worst</td>
</tr>
<tr>
<td>4 bit ripple adder</td>
<td>A=B=0000, Ci=0</td>
<td>91.3</td>
<td>Best</td>
</tr>
<tr>
<td></td>
<td>A=B=1111, Ci=1</td>
<td>94.0</td>
<td>Best</td>
</tr>
<tr>
<td></td>
<td>A=B=0101, Ci=1</td>
<td>282.9</td>
<td>Worst</td>
</tr>
</tbody>
</table>

Diagram:
- Primary Inputs
- Min-Leakage Vector
- Sleep
- Combinational Logic
Effectiveness of transistor stacking needs to be reevaluated as other leakage components become dominant. Optimizing a stack for subthreshold might lead to an increase of overall leakage due to gate tunnelling.

Patterns that optimize $I_{sub}$ and $I_{gate}$ are not the same!

Suboptimal choice! Best choice!

**BTBT current depends weakly on the input vector**
**Sleep transistors (MTCMOS)**

- **Forced stacking:**
  - Insertion of an extra series connected transistor (sleep transistor) in the pull-down/pull-up path of a gate
  - Sleep transistor switched OFF in the standby mode

- If the Vth of the sleep transistor is high, further leakage saving
- The drive current of forced stack is lower, hence increasing delay
  - Can be used for non-critical paths
Sleep transistors (MTCMOS)

**CHALLENGES**

- **Impact on area**
  the PMOS is usually very large since the “on” resistance is larger than that of the NMOS transistor.

- **Impact on performance**
  - Slow-down of power-gated logic cells when the circuit is active
  - Re-activation delay/energy for re-enabling powered down cells

- **Process modifications**
  for supporting the implementation of dual thresholds circuits

- **Design flow modifications**
  - insertion of sleep transistors

- **MTCMOS can only reduce power in standby mode**, while inducing an overhead in active mode

- when state cannot be lost in standby mode, need for high-Vth memory circuits to keep the state (keeper circuits)

- **Difficult to turn on and enforce the overdrive of the high-Vth sleep transistors with low supply voltages**
Optimized MTCMOS

• Instead of two sleep transistors, one can be used
• Usually NMOS:
  - $\mu_n > \mu_p \rightarrow$ smaller size
  - However, PMOS usually has a lower leakage

• One sleep transistor can be shared between several gates
  - Reduction in the number of sleep transistors, area overhead, dynamic and leakage power dissipations
  - Increase in the complexity of design optimization process
Design issues

• How many sleep transistors?
  - Affects the area overhead, the dynamic power overhead, and the leakage power saving

• How to cluster gates?
  - Affects routability and the size of the sleep transistors

• What size to choose for the sleep transistors?
  - Affects the delay and area overhead, the dynamic power overhead and the leakage power saving
**MTCMOS results**

- The MTCMOS techniques was applied to a 32-bit RISC processor used in a PDA
- Ground bounce: average=9mV, max=49mV
- Performance degradation = 2%

<table>
<thead>
<tr>
<th>Chip Size</th>
<th>Process</th>
<th># Gates</th>
<th>Clock</th>
<th>Total Sleep Transistors Width</th>
<th>Power Dissipation</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.7mm x 5.7mm</td>
<td>0.18µm 5-metal</td>
<td>1,914K</td>
<td>333MHz</td>
<td>18mm</td>
<td>270mW</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Leakage Power w/ MTCMOS</th>
<th>Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>2µW</td>
<td>6000x</td>
</tr>
</tbody>
</table>
Bouncing noise generated in one power-gating domain during a wake-up event is transferred through the shared power and ground distribution networks to the surrounding active circuit blocks. The node voltages and logic states of the active circuit blocks are thereby disturbed in a multi-domain MTCMOS circuit.
Adaptive Body Biasing (VTCMOS)

Variable Threshold CMOS is a body biasing design technique

In standby mode:
A deep reverse body bias (RBB) is applied to increase the threshold voltage and to cut off the leakage current

In active mode:
Forward body biasing (FBB) to achieve better current drive with less short-channel effects.

Area overhead:
- Logic cells with body bias contacts are up to 1.22x higher
- Otherwise the well is contacted by means of dedicated cells
  - Area overhead for well contacting cells
  - Well contacting cells placed at a certain distance to minimize $R_{bulk}$ resistance

In any case, Control loop circuitry and body bias distribution grid required
Triple well

Triple well process required for ABB!
ABB layout styles

(a) n-well
p-well
VSS
VPW
VNW
VDD

(b) Interval of body-bias straps

=Body-bias cell

=Area to place standard cells

= body contact
Body contact cells
VTCMOS Challenges

• Recent data shows a decreasing effectiveness of RBB to lower $I_{off}$ as technology scales due to the exponential increase in BTBT leakage due to halo doping in the source-substrate and drain-substrate junctions
  ✓ Reverse body bias should not exceed 500 mV to avoid tunnelling
  ✓ Range of threshold adjustment is limited

• Smaller channel lengths and lower channel dopings to reduce $V_{th}$ worsen the short channel effect, which in turn diminishes the body effect, and this in turn weakens the $V_{th}$ modulation capability of RBB

• Energy overhead for charging and discharging the substrate capacitance

• RBB worsens the short channel effect!

• Technology requirement: triple well!
  ✓ Not affordable for low-power, low-cost technologies

There is currently skepticism on the effectiveness of ABB in future technology nodes
Forward Body Biasing

It can improve circuit performance at lower short channel effects

HOW?
- Circuit designed with High-Vth transistors
  - leakage reduced in standby mode
- FBB used in active mode to boost performance

RESULT:

Both high channel doping and FBB reduce the short channel effect

With respect to low-Vth design, higher Ion for the same worst case Ioff

FBB can also be used as variability compensation technique
Case study: link variability compensation

- **2mm line model:**
  - $\pi$3 line scheme
  - RC values from BPTM website
  - Intermediate metal layer
  - 65nm technology

- **Full swing technique**
  - **Transmitter:**
    - Standard CMOS Flip-flop
    - Exponential horn driver
  - **Receiver**
    - Standard CMOS flip-flop

- **Low swing technique**
  - **Transmitter:**
    - Standard CMOS Filp-flop
    - NMOS push-pull driver
  - **Receiver**
    - Pseudo differential flip flop

**Same target performance for interchangeable links**
It is a clocked sense amplifier followed by a static latch

Uses single wire per bit while retaining advantages of differential signaling (low offset, high sensitivity)

Performance optimizations

- Substitution of PMOS P6 with NMOS N1
  - Faster evaluation

- Substitution of NOR Latch with static memory cell
  - Better input capacitance equalization
  - Better output transition time equalization
The link was then synthesized, placed and routed

- The link layout is highly irregular
  - 32 flits communication channel
  - VALID, STALL control signals were kept high-swing
  - Clock tree synthesis
  - 65nm STM technology
Crosstalk

- The clock strongly interferes with flit lines
- Full-swing technique still samples correctly
  - Low-swing technique can not cope with this high interference and the receiver fails
- Physical implementation of source synchronous synchronization techniques challenging
Both links work at the max. speed of full-swing (1.69 GHz); same clock propagation time
Swing reduction saves up to 80% of dynamic power
Low swing implies 28% lower leakage power
Low swing link area is only 1% bigger than the full swing one

PDIFF receiver: more area, less power
  - Due to some internal nodes switching at reduced swing
Nominal link design description:
- Hspice
- ST 65nm technology
- $\pi_3$ Model
- Channel length variation (WID)
  - Systematic
  - Random
Systematic variations

- The receiver propagation time (accounted from the clock edge to 50% output transition) changes adding systematic variation
- The transmitter is the most sensible component in the full swing link
- The receiver is the most sensible component in the low swing link
- The low swing is more robust to systematic variation than full swing
Random variations

- Random variations with $3\sigma/\mu=15\%$
- Full-swing: transmitter is the most sensitive component
- Low-swing: receiver is the most sensitive component (no malfunctioning with random variations below 20%)
- The low-swing shows a slightly better random variability robustness overall
Nominal link design description:
- Hspice
- 65nm ST technology

π3 Model

Channel length variation (WID)
- Systematic (5%)
- Random (15%)

Variability compensation techniques involved
- Adaptive Supply Voltage
- Adaptive Body Bias
Adaptive Supply Voltage (ASV)

- Charge pump
  - Provides variable supply from 1.0V to 1.2V with 0.1V step
    - Higher speed
    - Higher dynamic power
  - High efficiency ~90%

- Voltage island
  - Defines a limited chip area with the same supply voltage
  - The chip is divided in many islands
  - High tuning capacitance
  - High power cost
Adaptive Body Bias (ABB)

- **Forward Body Bias**
  - NMOS:
    - Not applicable here (no triple well technology)
  - Nwell (PMOS) body bias
    - Charge pump decreases the bias voltage from 1.0 to 0.5V with 0.1V step
    - Lower threshold voltage
      - Higher speed
      - Higher leakage
  - Low tuning capacitance
  - Low power cost

Supply voltage 1.0V
In the full swing link
- The ASV restores the nominal performance acting to the whole link requiring 23.5% extra power

In the low swing link
- The ASV restores the nominal performance acting only to the receiver requiring 8.5% extra power
With crosstalk

- In the full-swing link
  - Same as without crosstalk

- In the low-swing link
  - AVS has to increase the receiver supply voltage to 1.2V to restore the nominal performance therefore requiring 19.2% extra power (it was 8.5%)
ABB (on PMOS) efficiency

- **Fullswing link**
  - The ABB restores the nominal performance acting on the whole link requiring only 2.4% extra power

- **Lowswing link**
  - The ABB does not restore the nominal performance of all samples
  - Acting to the receiver provides almost the same performance than acting on the whole link
We verified that crosstalk and/or large random variations nullify ABB’s capability to restore unoperational samples in low-swing channels.
Adaptive voltage swing does not have the capability to restore the nominal performance of the low swing link. Increasing the swing voltage is highly power inefficient.
Summing up

- **The low swing link**
  - Similar performance wrt the full swing link
  - Higher power efficiency (5x less power)
  - Higher systematic variations robustness
  - Slightly higher random variations robustness
  - Good response to ASV compensation technique

- **ABB applied only to the PMOS provides a good compensation with**
  - Random variations below 20%
  - Full swing links