A System Delay Monitor Exploiting Automatic Cell-Based Design Flow and Post-Silicon Calibration

Hayate OKUHARA, Ryosuke KAZAMI, and Hideharu AMANO
Keio University, Japan
MCSoC 2019
Background

- Adaptive Voltage Scaling (AVS) has been an essential mean to achieve low power of VLSI systems
  - Dynamic power $\propto (V_{DD})^2$
  - Static Power $\propto \exp(\alpha V_{TH} + \beta V_{DD})$

- AVS efficiency has been endorsed by recent FD-SOI technologies
  - SOTB [1], UTBB [2], FDX [3]
  - Employing both an FD-SOI and AVS is imperative for recent IoT end-nodes

Trade-off between power and delay

- AVS has a trade-off between **Power** and **Delay**
  - $t_d = KV_{DD}/(V_{DD}-V_{TH})^\alpha$
  - Alpha power law [4]
- Lowering the power supply voltage too much incurs longer delay
  - The delay has to be less than the required performance
- Ways to find **appropriate voltages** are necessary

Conventional methodologies

- Estimating the worst-case delay under the process and temperature variations
  ✓ The obtained voltages are excessive for most of non-worst-case chips

- Integrating delay detection capabilities into FFs (e.g. Razor [5], Canary[6])

- Implementing replica circuits of critical path candidates [7]

- Conventional ways incur large overhead as various paths are monitored

Our proposal

• We propose a simple timing monitoring scheme for low-power VLSI systems
  ✓ Its circuit configuration is simplified as much as possible to reduce the power overhead
  ✓ The delay tracking capability is compensated by a post-silicon calibration

• The proposed monitor is fully implemented with a cell-based design automated flow
  ✓ Design cost is also an important concern, especially for low-cost SoCs
Proposed System Delay Monitor (SDM)

- The delay line emulates the entire system delay of a target system
  - Non-critical path candidates
  - Low power devices tend to have long paths resulting in large power overheads for their emulation

- $T_{\text{delay-line}}$ is adjustable via the configuration register
  - # of Delay units
  - NAND, INV, NOR, and the path to imitate wire delay behaviors

When $f_{\text{clk}} > 1/ T_{\text{delay-line}}$
This signal is asserted
Layout implementation flow

• A netlist of the SDM is firstly prepared
• It is integrated into a target system netlist
• The entire system is placed and routed (P&R)
  ✓ SDM is placed to a dedicated and fixed-size area

The interconnection path delay is automatically attached by the P&R.
Test chip

• The proposed SDM macro was implemented in a real chip
• SOTB 65-nm technology
• Design tools
  □ Synopsys Design Compiler
  □ Synopsys IC Compiler
• Power supply and body bias voltages (VBN VBP) are given from the outside of the chip
• Clock is also from the outside
• # of DU = 12.
• Each gate chain in the DU(INV NAND NOR) includes 10 cells
Frequency tracking capability (vs voltages)

- Same degree of body biases are supplied to both nMOS and pMOS
- The NAND and NOR delays are different from the INV chain
  - The NAND and NOR path can reduce $f_{\text{max-SDM}}$ while the INV path can increase $f_{\text{max-SDM}}$
Frequency tracking capability (vs others)

- \( T_{\text{delay-line}} \) has almost a linear dependency to \( \# \text{ of the used DUs} \)
- Interconnect delay is surely attached
How can we calibrate the SDM?

• $f_{\text{max-SDM}}$ has to always be lower than $f_{\text{max-targetsystem}}$

• VLSI System frequency is obtained by the continuous function
  \[ t_d = kV_{\text{DD}}/(V_{\text{DD}} - V_{\text{TH}})^\alpha \]

• The calibration is conducted at the two points (the slowest/fastest case)
  \[ \text{The SDM curve does not suddenly go above the target system curve due to the continuity} \]
Tested calibration algorithm

- Firstly, the operating condition is set to the fastest condition (e.g. vol. temp.)
- All of the DUs are set to the slowest path
  - The NAND path is the slowest
Tested calibration algorithm

- The path in the DU is changed to the smaller delay one
  - INV is the path of the smallest delay
- This procedure is repeated during $f_{\text{max-SDM}} < f_{\text{max-targetsystem}}$
Tested calibration algorithm

- The operating condition is set to the slowest condition
- If $f_{\text{max-SDM}} > f_{\text{max-targetsystem}}$, the small delay path in the DUs is changed to the larger one
  - NAND is selected
- This procedure is repeated to achieve $f_{\text{max-SDM}} < f_{\text{max-targetsystem}}$
The operating condition is set to the slowest condition

If \( f_{\text{max-SDM}} > f_{\text{max-targetsystem}} \), the small delay path in the DUs is changed to the larger one

- NAND is selected

This procedure is repeated to achieve \( f_{\text{max-SDM}} < f_{\text{max-targetsystem}} \)
Frequency tracking capability for a CNN accelerator

- $f_{\text{max-SDM}}$ is adjusted according to the $f_{\text{max-targetsys}}$ of a mW-range CNN accelerator [8]
- Body bias voltages are swept for the test (0.5V of RBB ~ 0.2V of FBB)
- 8.24% of tracking error, 1.67% of the power overhead

Conclusion and Future work

- A simple and low-overhead system delay monitor is proposed
  - It is implemented with a cell-based design automation flow
  - Real chip is fabricated with the SOTB 65-nm technology
  - Proposed SDM with a simple calibration scheme achieves several % of delay tracking error and a few % of the power overhead

- The used cells in the DUs should be optimized
  - What if AO, MUX, EXOR, etc, are used instead of INV NAND NOR?

- The current calibration algorithm does not fully utilize the SDM capability
  - How to exploit the interconnection path?

- The tested condition and target system are limited
  - The SDM should be also tested at wider voltage and temperature range with various target systems