Mini-Project 1

Leakage Power Reduction with Gate Sizing and Vt Cell Swap

Due: March 1, 2018, 11:59pm Pacific Time

Project Overview

- Gate Sizing

  Simultaneous **gate sizing** and **Vt cell swapping** to optimize circuit power under performance constraints

- Objective

  Minimize leakage power consumption under timing constraints

- Environments

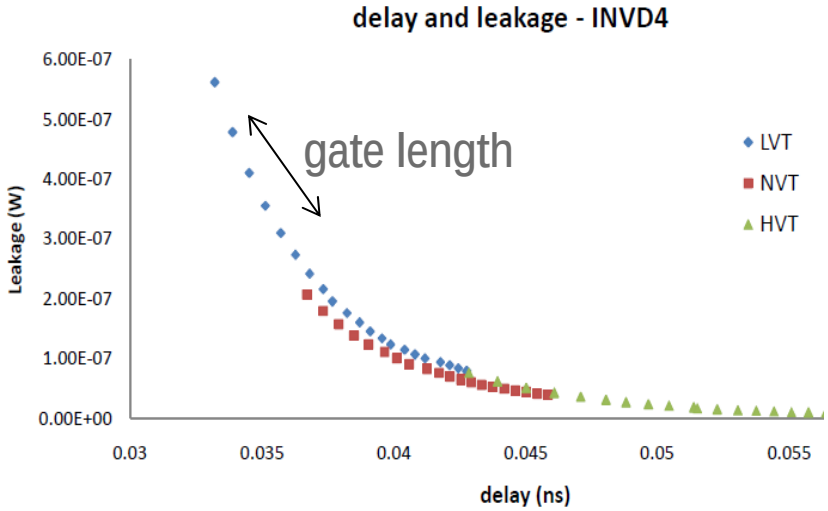
  Timing & power analysis: *Synopsys PrimeTime*

  Designs: usb_phy, aes_ciphe_top, mpeg2_top

  Library: Multi-Vt (HVT, NVT, LVT) library

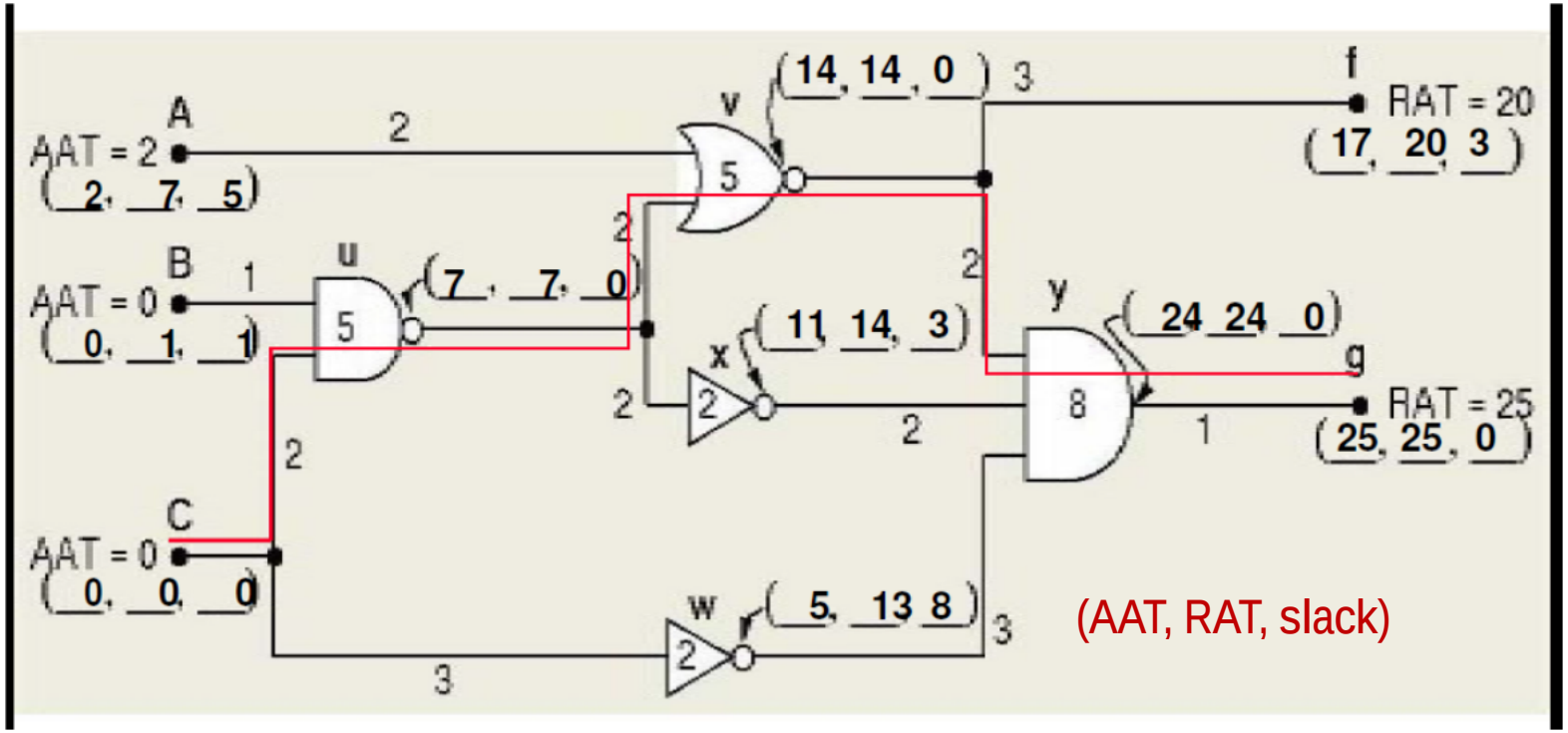
Background

- The sizing problem in VLSI design seeks to tune the circuit parameters (i.e., gate width, gate length and threshold voltage) to optimize a tradeoff of speed, area and power of circuit.
- Cell swapping with multi-Vt (threshold voltage) library is widely used method to reduce leakage power in post-layout stage
- Large size & LVT : fast timing, but large leakage
- Small size & HVT: slow timing, but low leakage



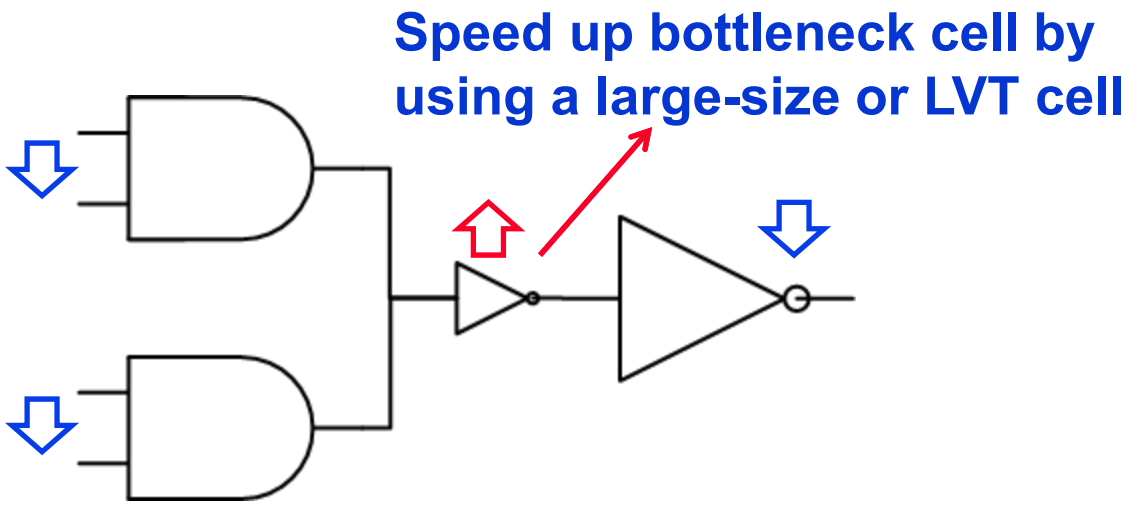
- Use low-leakage and/or small-size cells along paths with positive timing slack

Cell Swap Example



- Assume that all cells are NVT (nominal Vt) type
- Cell “x” and “w” can be swapped to a HVT or small-size cell since they have positive slack (3 and 8)
- In case, slack of “v”, “x”, “w” are zero, down sizing or using HVT cells can lead to timing violation

Timing Recovery Can Help



- Speed up bottleneck cells that participate in many timing-critical paths
- With timing recovery, we are able to down size or use low leakage cells on critical paths to optimize power

Environment

- Design: 2013 ISPD benchmarks
  - ISPD Liberty + TSMC 65nm BEOL
  - P&R is implemented with Cadence Innovus Implementation System
- Liberty: 2013 ISPD
  - Vt types: triple-Vt (HVT, NVT, LVT)
  - Cell sizes: (except flip-flop)
- Timing constraints each cell has 10 sizing variants
  - Max transition constraints (described in Liberty)
  - Max capacitance constraints (described in Liberty)
  - Setup time constraint

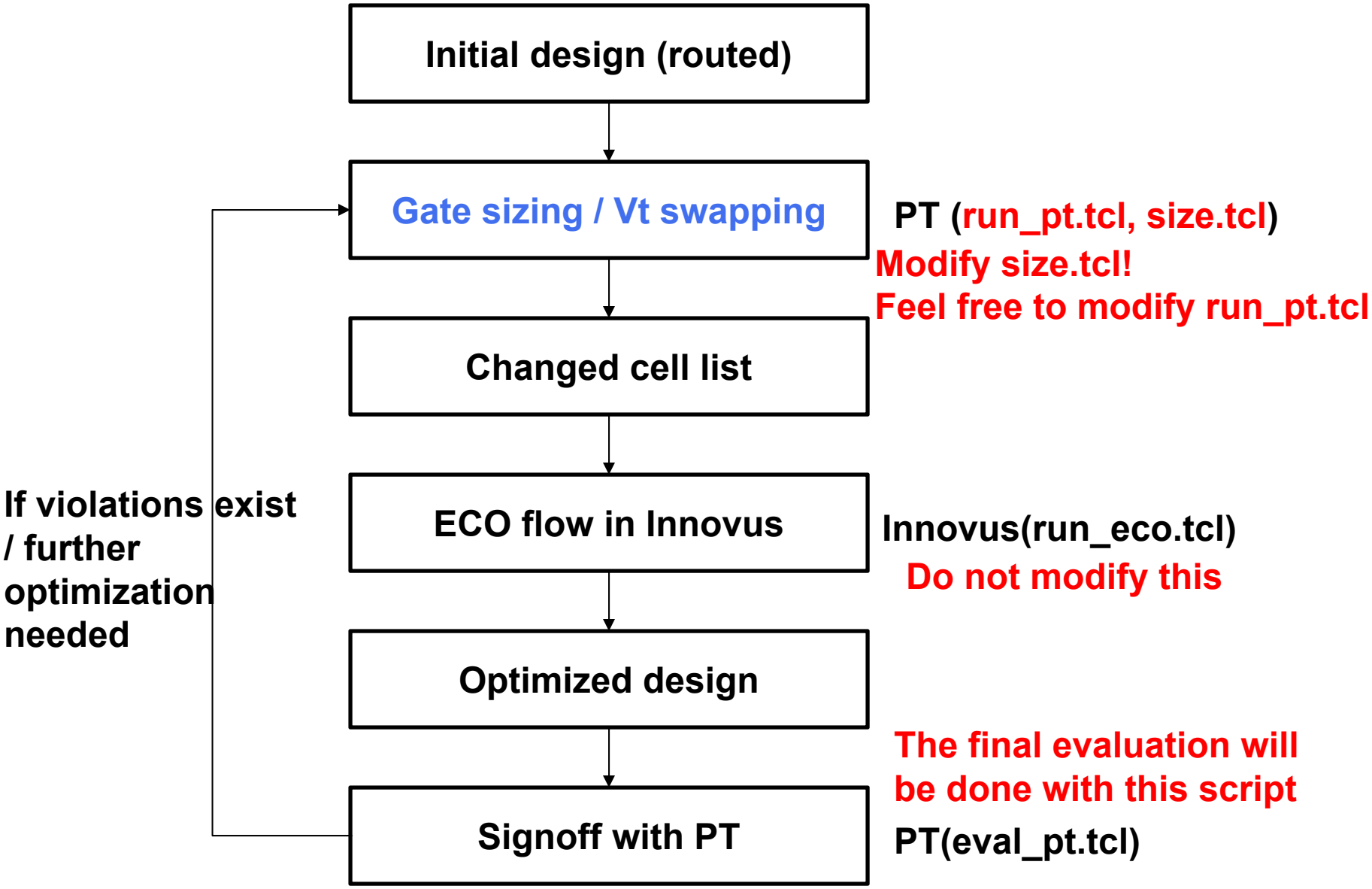
Library Information

Cell sizes

Cell name in benchmarks	Mapped cell type
in01{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	Inverter
na02{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	2-input NAND
na03{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	3-input NAND
na04{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	4-input NAND
no02{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	2-input NOR
no03{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	3-input NOR
no04{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	4-input NOR
ao12{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	2-1 AOI
ao22{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	2-2 AOI
oa12{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	2-1 OAI
oa22{s, m, f}{01, 02, 03, 04, 06, 08, 10, 20, 40 ,80}	2-2 OAI
ms00f80	D Flip-Flop

Vt types: s = HVT, m = NVT and f = LVT

Overall Flow



PrimeTime Execution

- Running PrimeTime

```
ieng6-ece-01$ pt_shell –f run_pt.tcl
```

- Report after sizing / Vt swapping

${design}_sizing.rpt

```
Initial slack:  0.010831 ns
Initial lekeage:       2.81879e-05 W
=======================================
- cell FE_RC_1237_0 is swapped to oa12s04
- cell FE_RC_1141_0 is swapped to na02s02
.........
=======================================
Final slack:    0.000083 ns
Final lekeage:  2.50146e-05 W
#cell swaps:    160
Leakage improvment      11.258 %
```

PrimeTime Baseline Script (size.tcl)

```
set cellList [get_cell *]                                          # put all cells into cellList

foreach_in_collection cell $cellList {
    set cellName [get_attri $cell base_name]                       # get cell instance name
    set libcell [get_lib_cells -of_objects $cellName]
    set libcellName [get_attri $libcell base_name]                 # get library cell name

    if { [regexp {[a-z][a-z][0-9]f[0-9]f[0-9][0-9]} $libcellName] } { # change LVT to NVT
        set newlibcellName [string replace $libcellName 4 4 m]
        size_cell $cellName $newlibcellName

        set newWNS [ PtWorstSlack clk ]                            # calculate worst slack
        if { $newWNS < 0.0 } {                                     # restore the swap
            size_cell $cellName $libcellName                       # if WNS goes to negative
    }
}
```

- You are supposed to modify this script to enable sizing and Vt assignment to reduce leakage more
- **You should not use PT's own optimization commands (e.g., fix_eco*)**
- **Your solutions should be reproducible by your script (If you performed manual changes, please list all of them and explain why you did this.)**

Potential Knobs for Your Optimization

- See Zahn's SNUG'08 paper

  http://vlsicad.ucsd.edu/SIZING/ref/other/ZahnSNUG08.pdf

- Sensitivity based greedy method

  Sort cells according to the sensitivity and try cell swaps.

  Prof. Kahng's papers :
  http://vlsicad.ucsd.edu/SIZING/ref/our/GuptaKSS06.pdf

  http://vlsicad.ucsd.edu/Publications/Conferences/288/c288.pdf

  http://vlsicad.ucsd.edu/Publications/Conferences/304/c304.pdf