



ORACLE®

A Tool for Exploring Advanced RTL Clock Gating Opportunities in Microprocessor Design

*Krishnan Sundaresan (krishnan.sundaresan@IEEE.org),
Aravind Oommen, Doug Meserve, Hemanga Das,
Jaewon Oh, & Mohd Jamil Mohd*

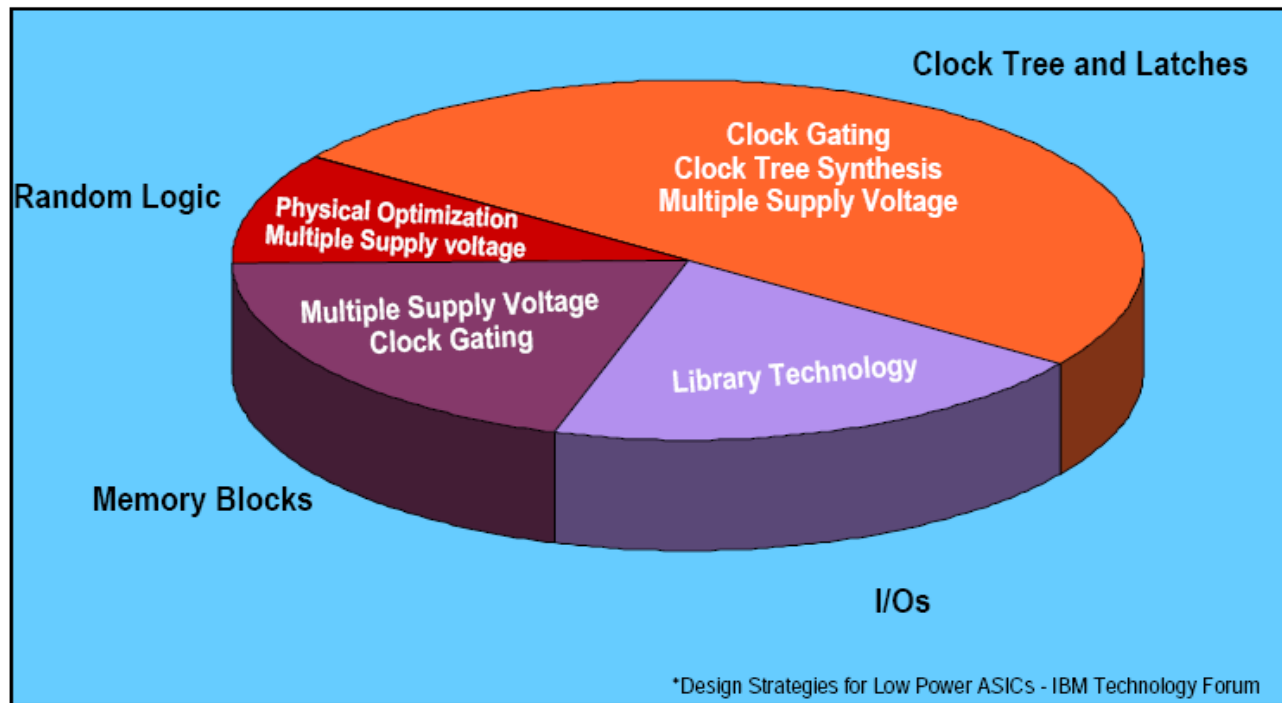
Oracle America, Inc.

Outline

- Power reduction and clock gating (CG)
- Our new methods and clock gating tool
- Tool features for RTL CG exploration
- Results
- Summary and discussion

Active Power Reduction Techniques

- Active power is still a dominant chunk of total power dissipation in high-performance microprocessor designs
- Several techniques have been used to address active power reduction in circuits



Power Reduction and Clock Gating

- Power dissipation in flops and clock lines are a substantial portion of total active power (70% in IBM Power5 [1], 60% in Sun UltraSPARC T2 [2])
 - Much of this power dissipated when clock signals are switching even when data is not
- Fine-grained clock gating (CG): disable clock to flops/registers when the information they hold is not used in the subsequent stages
 - Can save up to 30% of total power [3]
 - Savings are obtained in
 - Flop power when not in use
 - L1clk (leaf header to flop) net power
 - Some MUXes/gates in feedback paths can be removed

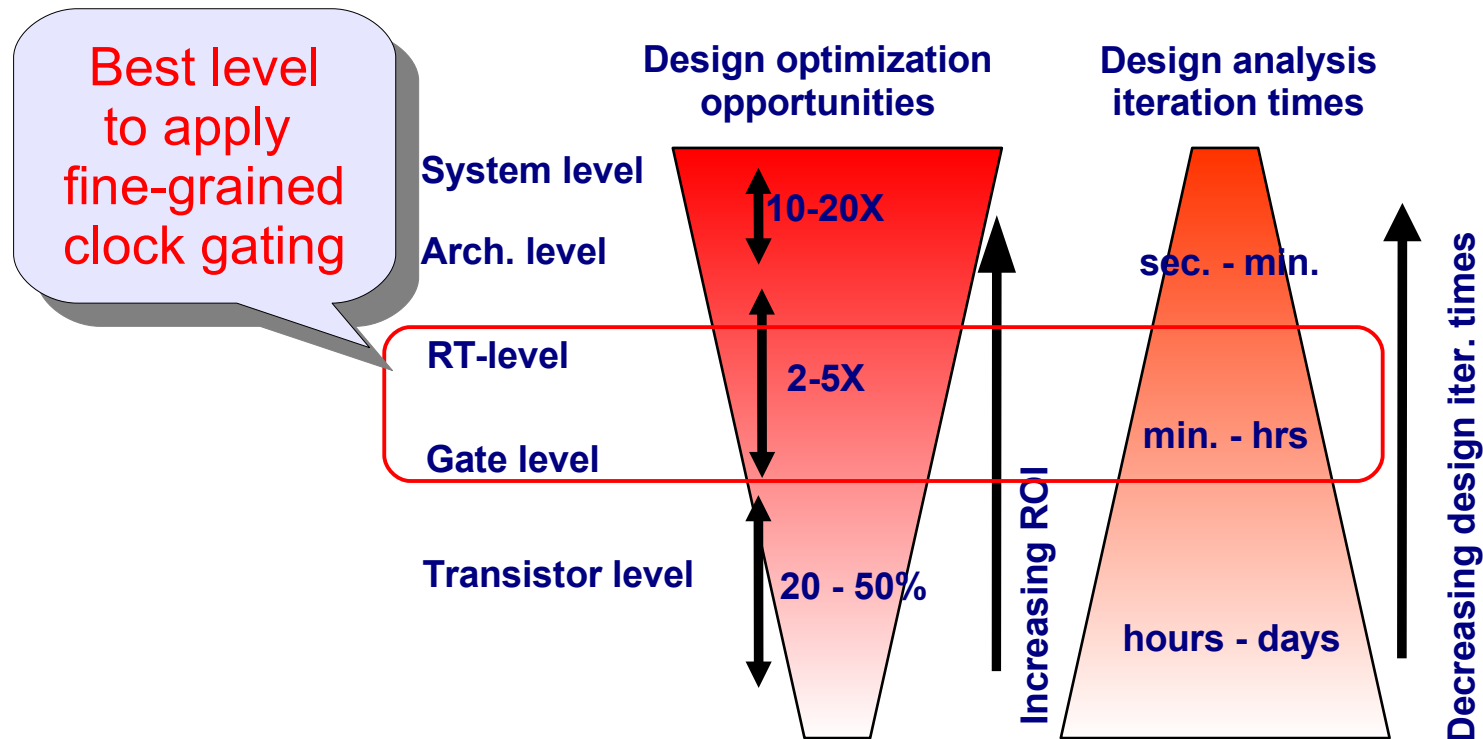
[1] Jacobson *et al.*, Proc. HPCA 04

[2] Internal estimate

[3] Bose *et al.*, Hotchips 2008

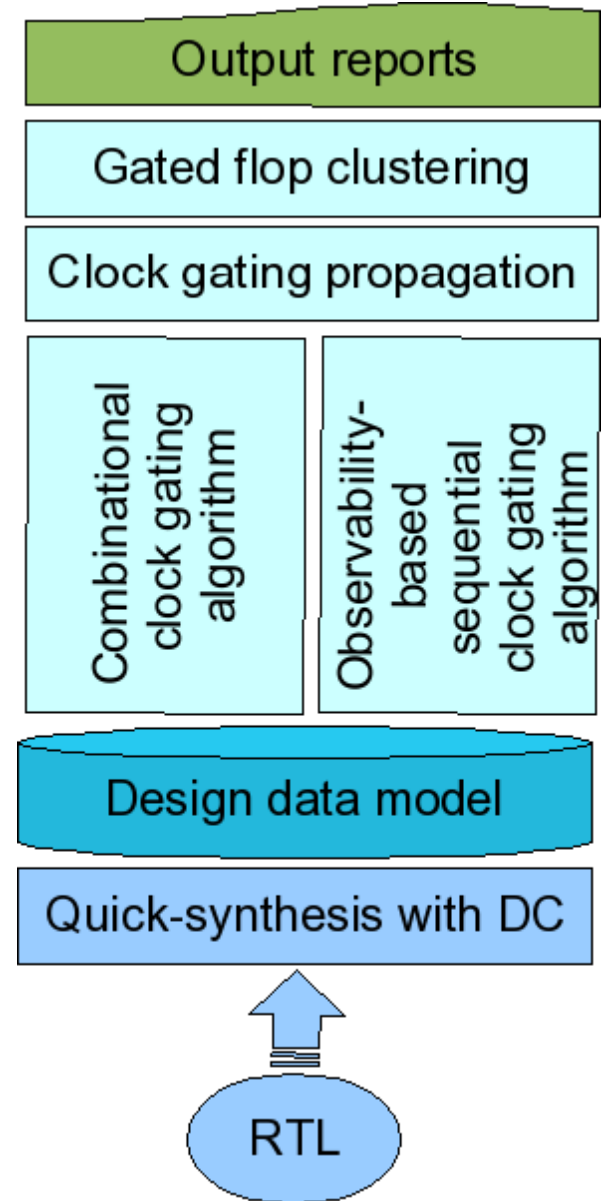
Clock Gating in the Design Flow

- Power optimizations done earlier in the design cycle result in higher returns on investment. Design iteration times are also significantly faster in early stages



Overview of Our Tool

- Read behavioral RTL and quick-synthesize (via Synopsys Design Compiler)
- Read and store the design in our custom data model
- Identify advanced combinational and sequential clock gating opportunities, propagate clock gating across the design, and create clock-gated flop clusters using our proprietary algorithms (implemented in C++)
- Generate reports of gatable flops, their enable signals in RTL namespace, and CG efficiency (power saving metric)



Features of Our Tool

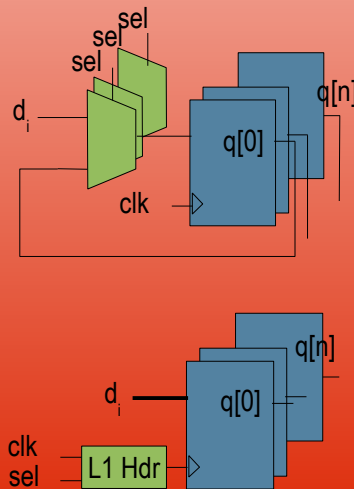
- Solution that implements advanced clock gating methods and CG verification techniques developed internally at Oracle, with the aim of achieving best power reduction and implementation trade-offs
- Seamless integration into our front-end design flows and other in-house CAD tools for microprocessor design
- Two built-in verification options for CG: simulation-based and Boolean equation based
- Clock gated netlist write-out

Features Facilitating CG Exploration

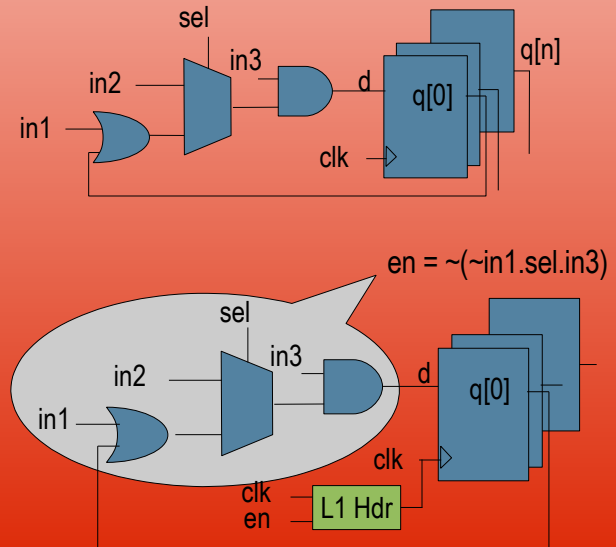
- Provides advisory reports listing all the CG opportunities in the design, and most efficient way to group & implement them
 - User can explore different ways of clustering clock gated flops based on min/max flops per cluster
 - Clock gating efficiency is weighted on a per-cluster basis. Groups of flops that provide the best power reduction can be easily identified and clock gated first
- Identifies already existing gating in given RTL and can expand/re-use it to clock gate more flops
- Uses waveform dump (FSDB or VPD) to estimate clock gating efficiency. More than one dump can be used simultaneously

Combinational Clock Gating

- Single-gate feedback CG
 - Find & gate Mux+Flop (holding) registers by mapping such elements to a custom “dummy” clock-enabled flop during quick synthesis

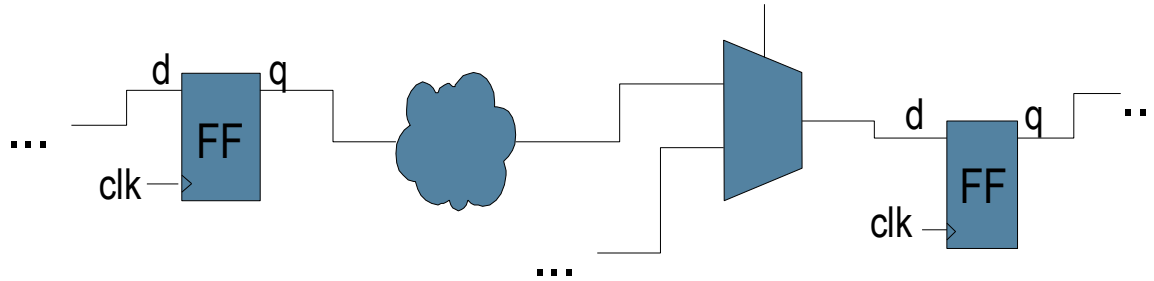


- Multi-gate feedback CG
 - Find flops with feedback loops containing multiple gates and find clock disable condition corresponding to cycles when the feedback is ON



Sequential Clock Gating

- Some designs have almost no feedback paths (e.g.) mux-pipelined datapaths

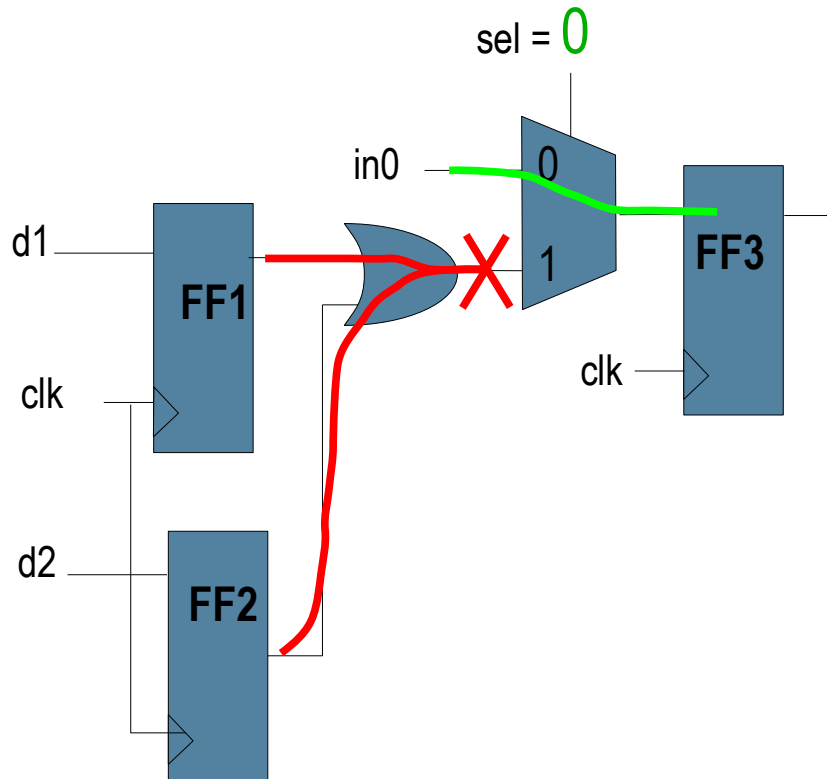


- Limited opportunities for combinational CG exist in these
- Our tool uses “observability-based” technique to find CG [4]

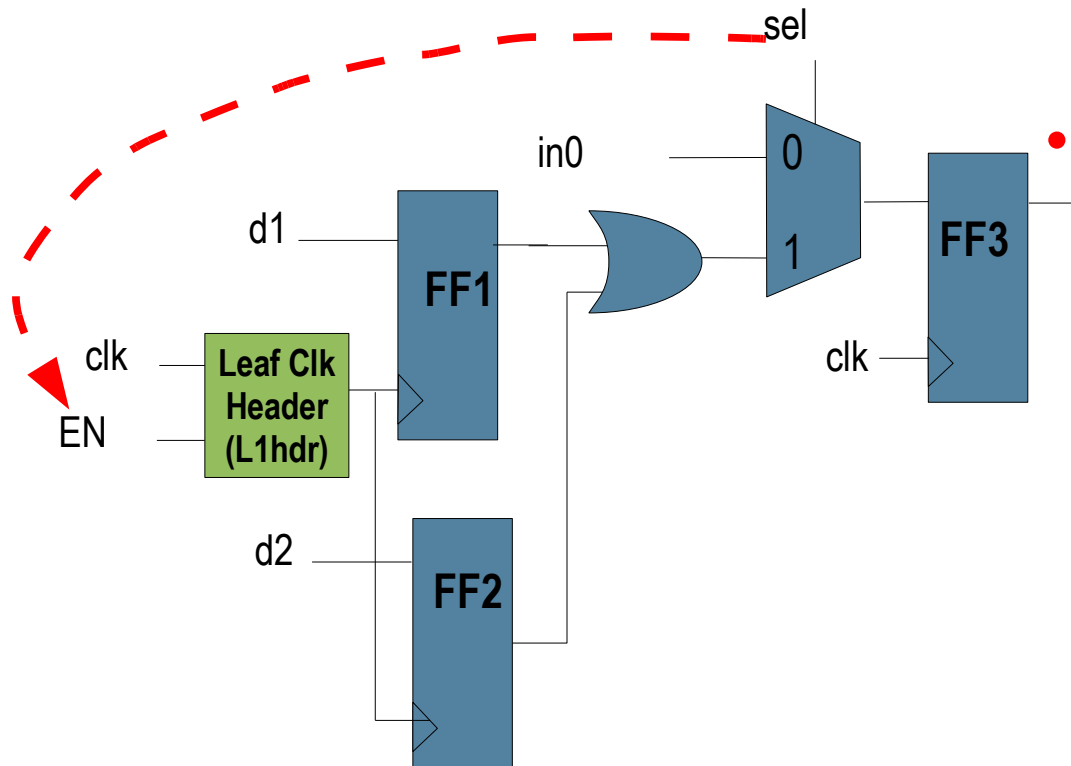
[4] Damiani *et al.*, ICCAD, 1990.

Observability-Based Clock Gating

- FF1 and FF2 are not observable when sel is low



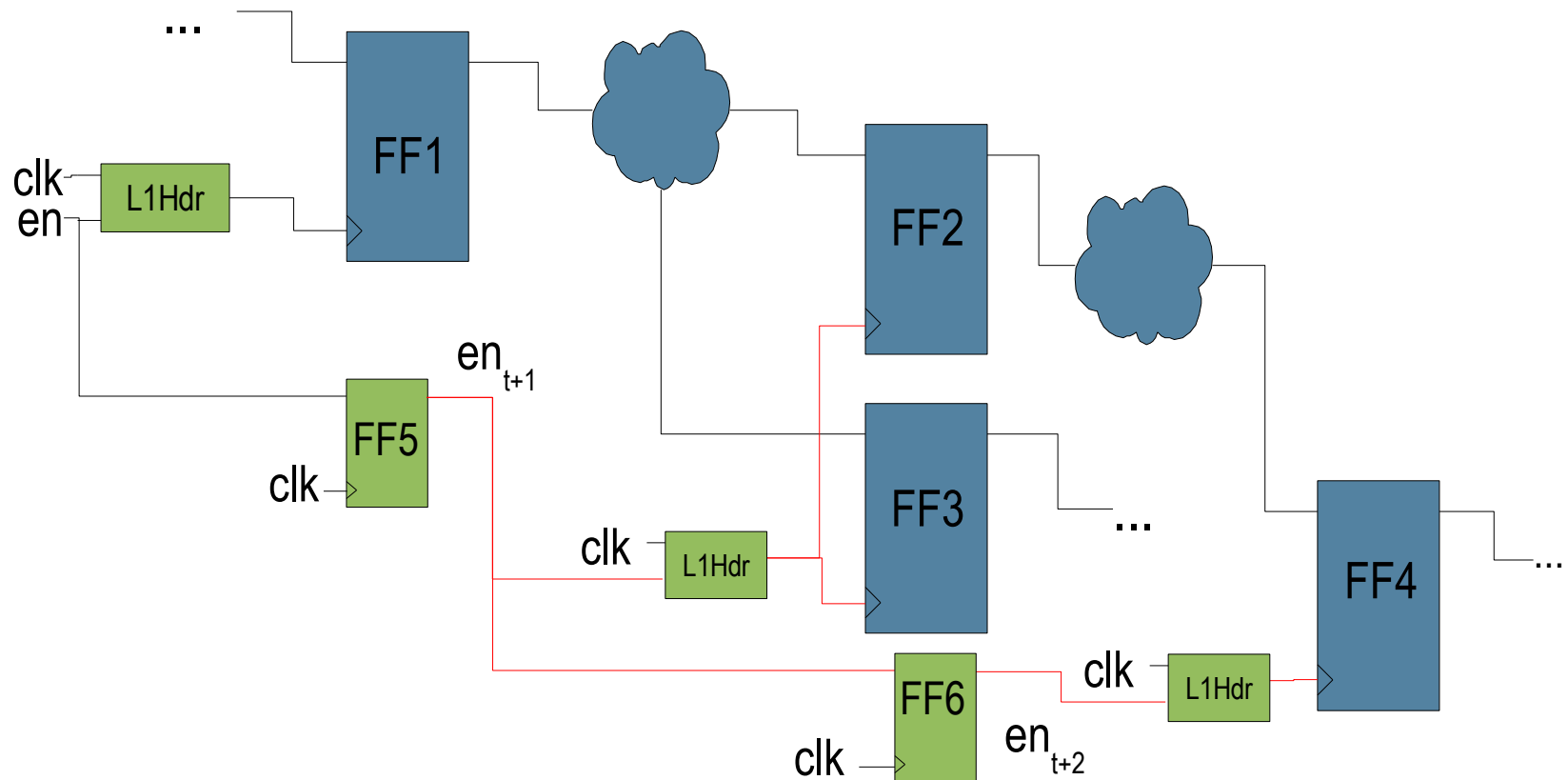
Observability-Based Clock Gating



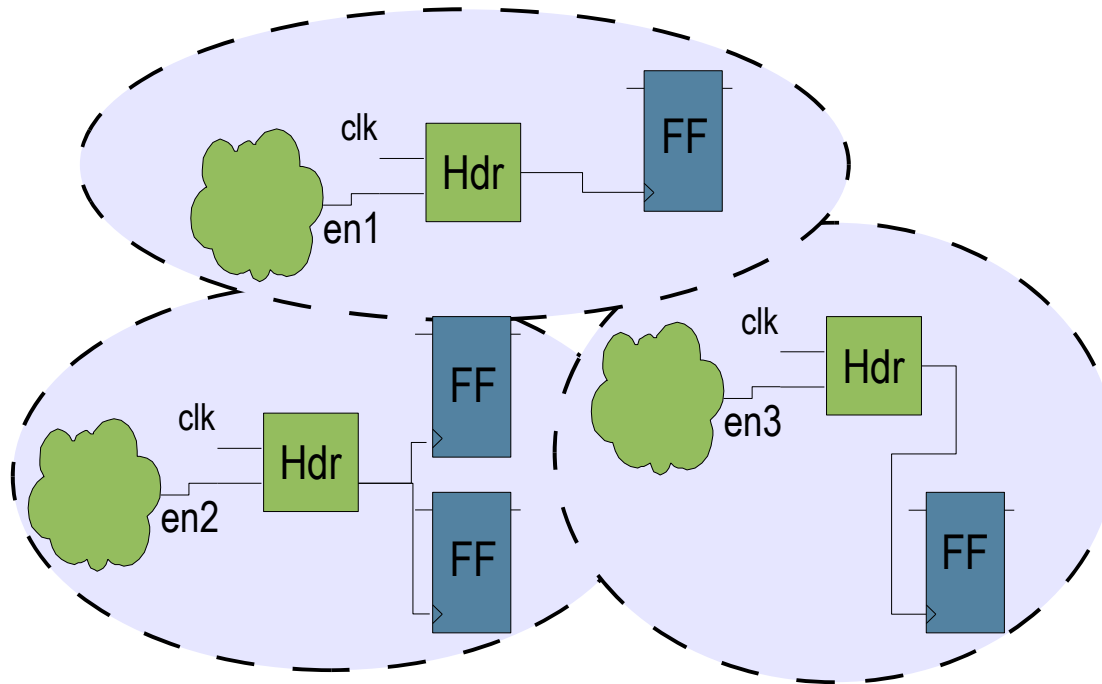
- FF1 and FF2 are not observable when sel is low
- sel can be used as clock gating enable (EN) for FF1 & FF2
 - EN = sel, obtained one clock cycle earlier

Clock Gating Propagation

- Using the clock gating enable found for flops upstream, free-running flops downstream can be clock gated using staged versions of the same enable



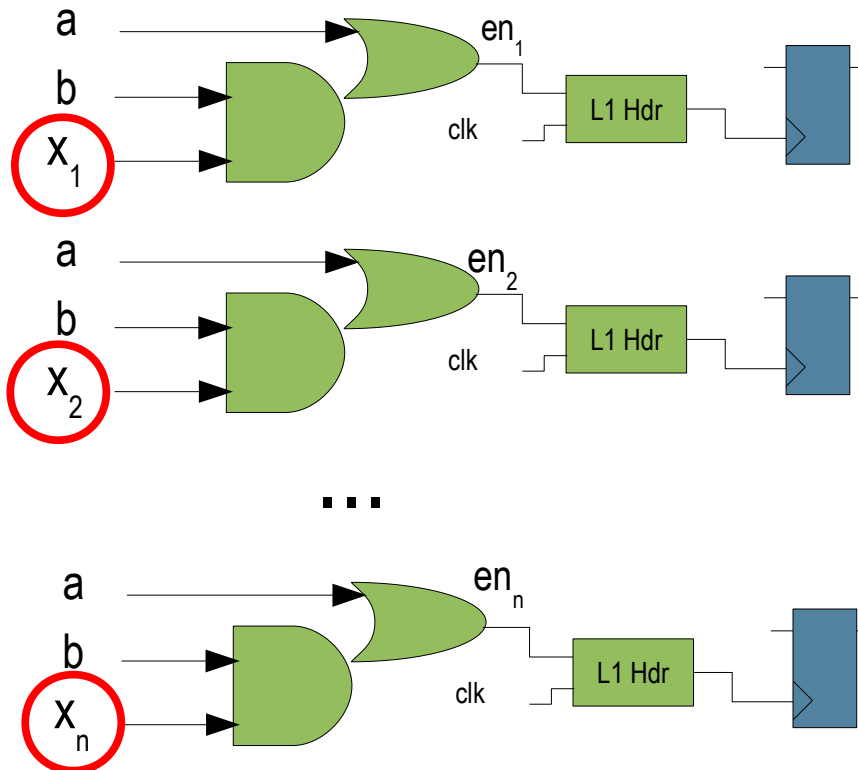
Enable Consolidation Technique



- Usually have many “leftover” enables not shared by enough flops
 - CG implementation complexity would be too high
 - Likely no net power reduction
- How to make these usable?

Enable Consolidation

$$en_i = a + x_i \cdot b$$



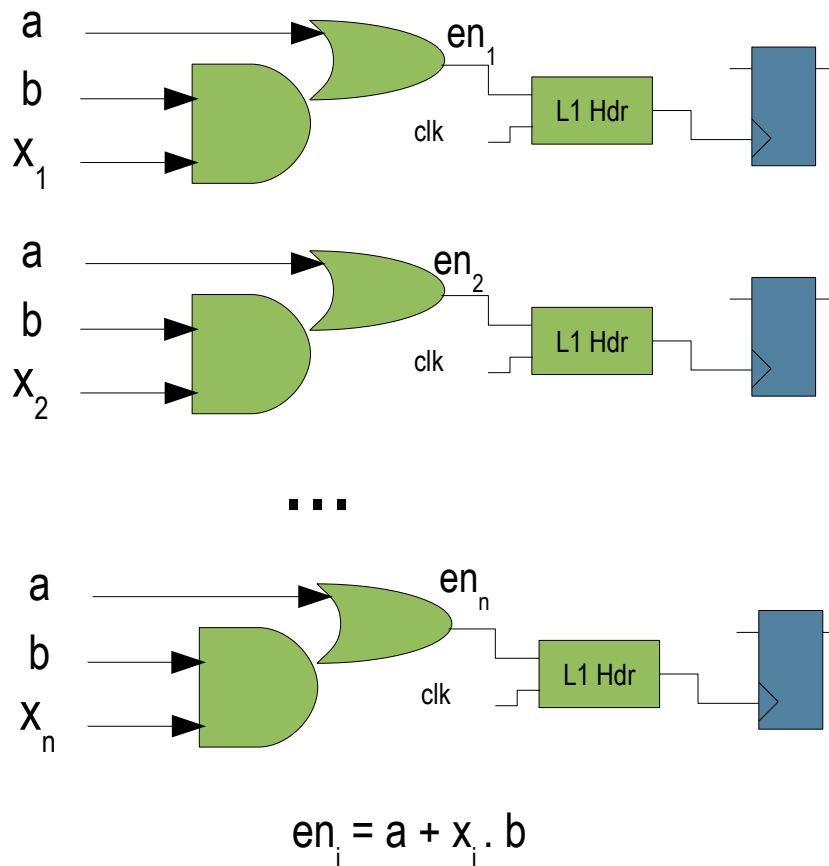
- Observation: only minor differences between these enable equations
- Can trim variables from these equations and combine them in a safe manner using *existential quantification* [5]

[5] Weaver *et al.*, JSAT, 2006

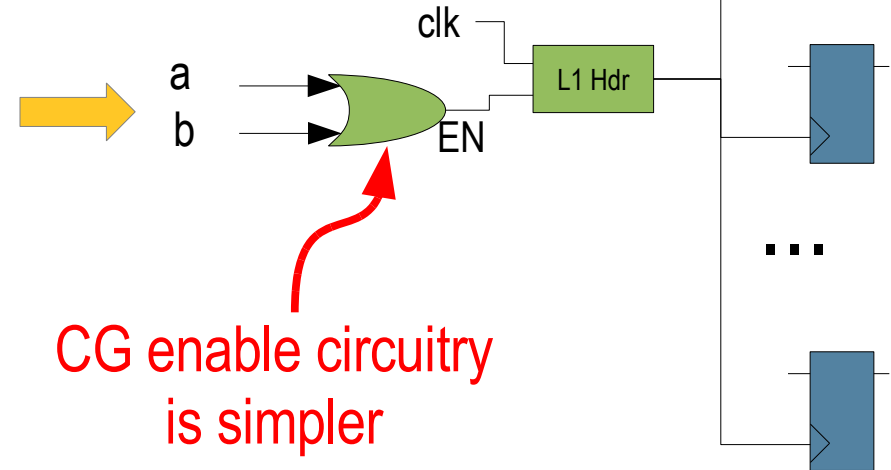
Enable Consolidation

a	b	x_i	en_i	EN
0	0	0	0	0
0	0	1	0	0
0	1	0	0	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Sacrifices some chances
to shut off clock



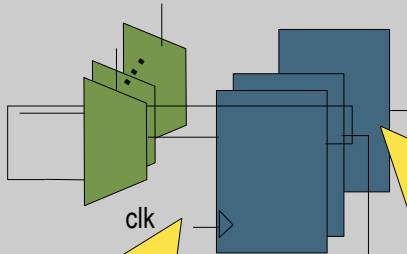
More flops share
same enable



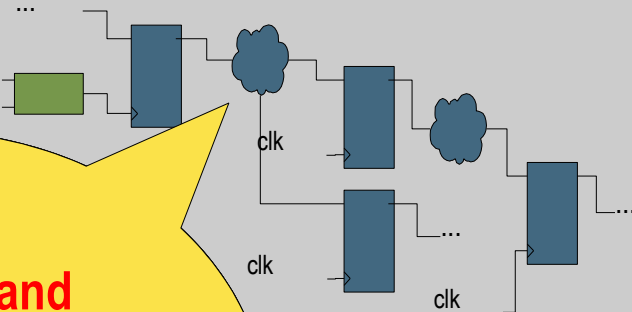
CG enable circuitry
is simpler

Validating CG Results from Our Tool

Single-Gate Feedback



CG Propagation

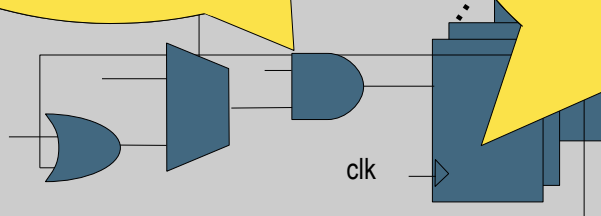


**Formal
(Verplex/LEC)**

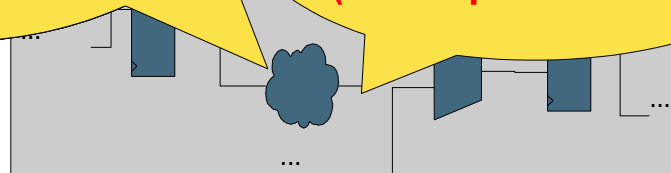
**Simulation and
waveform comparisons
to check block I/O
behavior does not
change**

**Additional BDD-
based technique
(developed internally)**

Multi-Gate Feedback



Observability Based



Results: CG Coverage on Ungated RTL

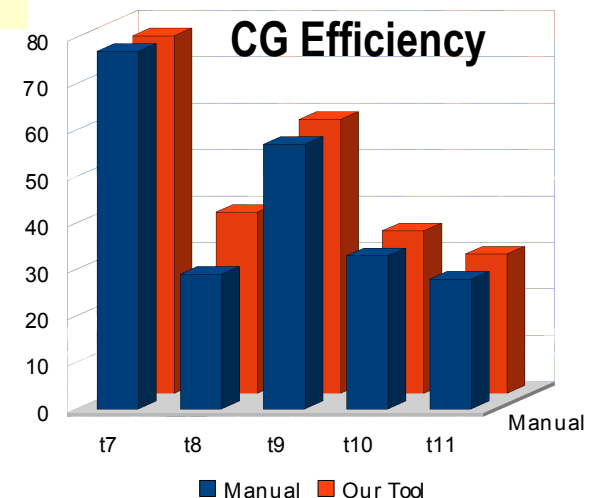
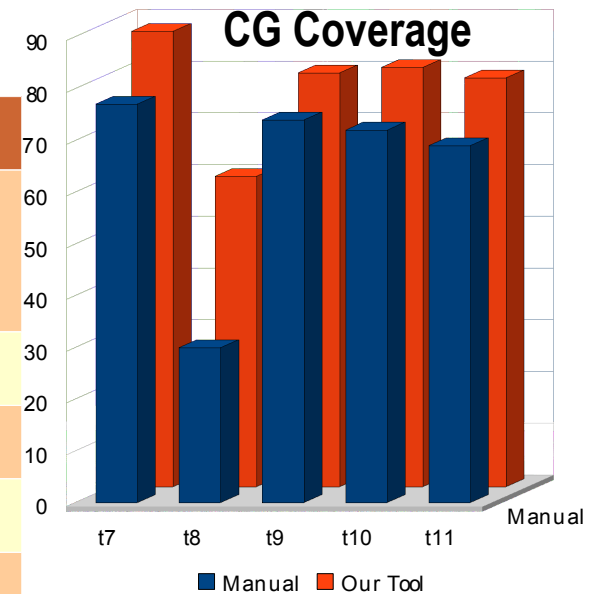
Design	# Flops	Our Tool
t1	1495	39%
t2	1610	50%
t3	2851	46%
t4	8420	27%
t5	8959	37%
t6	17218	86%

- Our tool found significant CG coverage in RTL (UltraSPARC processor blocks) that was not clock gated in a prior generation design

Results: CG Coverage & CG Efficiency on Gated RTL

Design	# Flops	Coverage		Efficiency	
		Previous design	Our Tool	Previous design	Our Tool
t7	2752	77%	88%	77%	77%
t8	3586	30%	60%	29%	39%
t9	5248	74%	80%	57%	59%
t10	7166	72%	81%	33%	35%
t11	8975	69%	79%	28%	30%

- Our tool was able to improve clock gating coverage and efficiency on many designs that already had clock gating
- CG Efficiency is %-age number of cycles the clock to the flops is turned off. Used as a metric to estimate power savings.



Summary

- Developed an in-house clock gating tool to facilitate RTL clock gating exploration
- Implemented state-of-the-art and Oracle-proprietary algorithms for finding combinational and sequential clock gating in the tool, yielding best clock gating coverage and efficiency for our advanced microprocessor design needs

Acknowledgment

- Walid Elgharbawy
- Wei-Lun Hung
- Michel Laudes
- Rob Mains
- Sujeeth Udipi
- and, all our tool users.

References

- Jacobson H. et al., *Stretching the limits of clock gating in server-class processors*, in Proceedings of International Symposium on High-performance Computer Architecture (HPCA-11), San Francisco, CA, Feb. 2005.
- P. Bose et al., *Power-Performance comparative evaluation of alternative microarchitectures*, HotChips-2008.
- Fabio Somenzi, CUDD: CU Decision Diagram Package, <http://vlsi.colorado.edu/~fabio/CUDD>
- M. Damiani and G. De Micheli, *Observability don't care sets and Boolean relations*, Proceedings ICCAD, 1990.
- Weaver S. et al., *Extending existential quantification in conjunctions of BDDs*, Journal on Satisfiability, Boolean Modeling, and Computation, volume 1, pp. 89-110, 2006.



Q & A