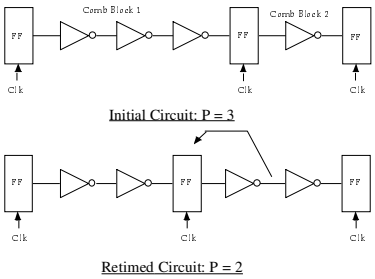
Fast Algorithms for Retiming Large VLSI Circuits

Sachin S. Sapatnekar
Department of ECE
University of Minnesota
Minneapolis, MN 55455

Outline of the talk

- **Introduction**
- Retiming and clock skew
- Minperiod retiming
- Minarea retiming
- Conclusion

What is retiming? (in pictures)



What is retiming? (in English)

- Relocation of memory elements (to achieve lower clock periods/area/power etc.)
- Maintain the latency of all paths in circuit
- Idea proposed by Leiserson and Saxe (1983, 1991), with an $O(|G|^3)$ algorithm
- Contribution of this work:
  - new theory and algorithms for large circuits
  - fast implementations (> 50K gates in minutes)

Related work

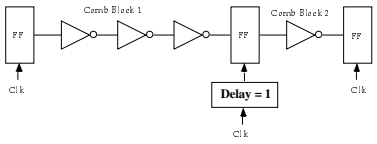
- Leiserson and Saxe (Algorithmica 1991)
  - Presented theory and algorithms for retiming
- Shenoy and Rudell (ICCAD 94)
  - Practical implementation of Leiserson-Saxe
- Fishburn (IEEE Tr Computers July 90)
  - Concept of clock skew optimization using linear programming

Outline of the talk

- Introduction
- **Retiming and clock skew**
- Minperiod retiming
- Minarea retiming
- Conclusion

Deliberate clock skews

- In a synchronous circuit, all clock signals arrive at all FF's at exactly the same time
- If the clock signal at an FF is delayed?

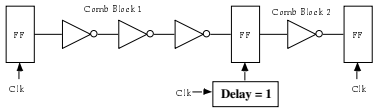


Clock skews

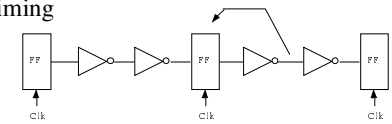
- Can use clock skews for deliberate cycle borrowing!
- Relation between retiming and skew?

Retiming and skew

- Skew



- Retiming



Outline of the talk

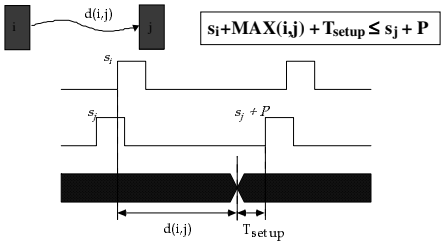
- Introduction
- Retiming and clock skew
- **Minperiod retiming**
- Minarea retiming
- Conclusion

[As Archimedes said in his bathtub...

- New algorithm (Deokar and Sapatnekar TCAD 10/96)
  - Phase A: Find optimal skews
    (complexity depends on the number of FF's and **not** the number of gates)
  - Phase B: Relocate FF's to retime circuit
    (since most FF movements are seen to be local in practice, this should not take too long)

Zero-clocking constraints

- Data must not reach destination FF too late



$$s_i + MAX(i,j) + T_{setup} \leq s_j + P$$

Optimal clock period

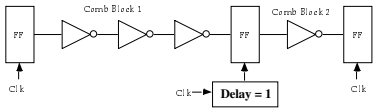
- Represented by the optimization problem below (Fishburn 90) - solve for P, s_i values

minimize P
subject to
(for all pairs (i,j) of FF's)
Zero Clocking: $s_i + MAX(i,j) + T_{setup} \leq s_j + P$

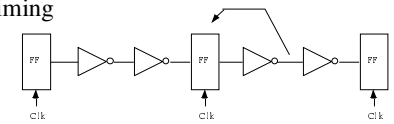
- MAX(i,j) constant ⇒ difference constraints
- Efficient graph-based solutions

Retiming and skew

- Skew

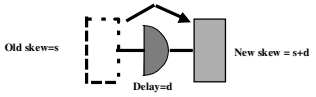


- Retiming



From skews to retiming..

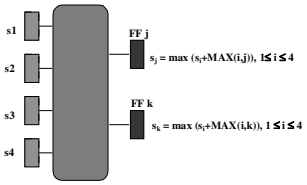
- Moving a flip-flop across a gate G
  - left->right ≡ increasing its skew by delay(G)
    ⇒ move negative skew FF's to the right



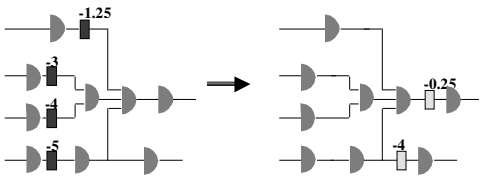
  - right->left ≡ reducing its skew by delay(G)
    ⇒ move positive skew FF's to the left

More generally...

- Moving FF's across a comb. subcircuit left->right (right->left analogous)



Example



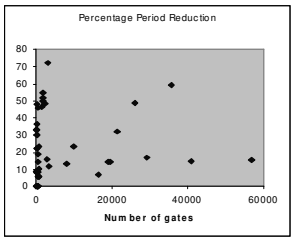
Theoretical results

- Lemma
  - At the end of the retiming procedure, the skew at each FF is no more than half a gate delay
- Theorem
  - If all skews are set to zero at the end of the retiming procedure, then the clock period for the circuit is no worse than $P_{opt} + d_{max}$
    where P_{opt} : optimal clock period using skews
    d_{max} : largest delay of any gate in the circuit.

Experimental results

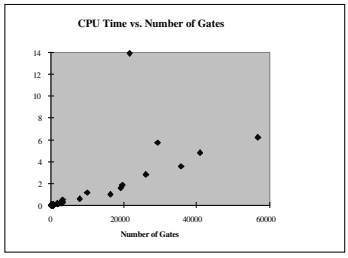
- Algorithm implemented as ASTRA
  (A Skew-To-Retiming Algorithm)
- Tested on ISCAS89 benchmark suite
- Largest circuit (> 50000 gates) retimed in well under a minute (Sun 1/170)
- Assumed unit delay gates; found that
  - $P = P_{opt}$ when integral
  - $P = \lceil P_{opt} \rceil$ when fractional.

Percentage period reduction



$$\text{Percentage reduction} = \frac{(P_{original} - P_{retimed}) \times 100}{P_{original}}$$

Execution times



Problem!

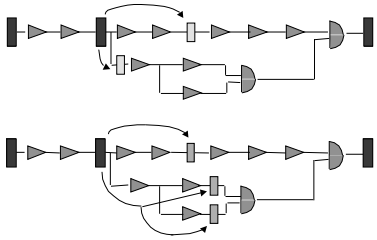
- Minperiod retiming pays no regard to the number of FF's!
- #FF's can become very large! (up to 4x of the optimal number)

Outline of the talk

- Introduction
- Retiming and clock skew
- Minperiod retiming
- **Minarea retiming**
- Conclusion

Minarea retiming

- Need to minimize the number of FF's



[Quick] Outline of Leiserson-Saxe

- Terminology
  - $r(v)$ = # FF's that move R-to-L across a gate
  - $w(u,v)$ = # FF's on an edge between u and v
- Constraints
  - For each edge between gate u and gate v

    $w(u,v) + r(v) - r(u) \geq 0$ (Nonnegativity)
  - For any path between gate u and gate v with delay > P and total weight $W(u,v)$,

    $W(u,v) + r(v) - r(u) \geq 1$ (Period constraint)

Minarea retiming - L&S

- Basic idea:
  - Leiserson&Saxe showed that minarea retiming is an LP which is the dual of a minimum cost network flow problem
  - Solve for r variables for each gate
  - Number of constraints can be extremely large
    - » Pruning methods suggested in L&S paper, implemented in Shenoy/Rudell with a fast constraint generation technique

Our approach - Minaret

- Leiserson-Saxe:
  - # variables = $|G|$, # constraints = $O(|G|^2)$
- Our approach*
  - Prunes # variables (23K to 20K - s38584.1) and # constraints (11M to 97K - s38584.1)
  - Generates the constraints more efficiently
  - LP solved in minutes by network simplex

    (8 sec for s38584.1)

* Maheshwari and Sapatnekar, DAC 97

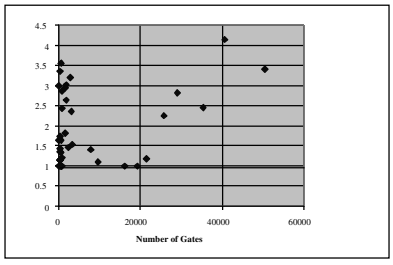
Constraint reduction (outline)

- Consider the constraint

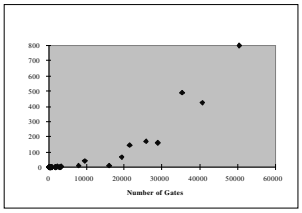
  $r(u) - r(v) \leq c_{uv}$
  - Given $L_u \leq r(u) \leq U_u$ and $L_v \leq r(v) \leq U_v$

    $r(u) - r(v) \leq U_u - L_v$
  - If $U_u - L_v \leq c_{uv}$: constraint implied by bounds
  - Can also reduce mirror vertex constraints
  - Constraints from/to fixed vertices redundant

(# FF's: (#Minperiod)/(#Minarea)



CPU time for Minaret



(Previous best: 39 hours for a 10K gate circuit)

Outline of the talk

- Introduction
- Retiming and clock skew
- Minperiod retiming
- Minarea retiming
- **Conclusion**

Retiming level-clocked circuits (outline only!)

- Harder problem
  - Critical path between two gates may change with clock period (Lockyear/Ebeling TCAD 9/94)
  - Cannot stop constraint generation after encountering one latch due to existence of multicycle paths
- Retiming latches fits in very well with skew idea (picture a latch as FF with some allowable skew)
- Fast algorithms for minperiod/minarea retiming of large developed* (> 50K circuit in 18 sec/80 min)

* Maheshwari and Sapatnekar, ICCD96, TAU97

Conclusion

- Fast algorithms for minarea/minperiod retiming of large edge-triggered circuits
- Extended to the harder problem of retiming level-clocked circuits
- Acknowledgements:
  - Naresh Maheshwari, Rahul Deokar (students)
  - Jack Fishburn, Bell Labs (discussions)

REAL! Conclusion

Thank you for your attention

I would be glad to answer any questions