Announcements

- Homework 3 posted this week, due after Spring break
- Quiz #2 today
- Midterm project report due on Wednesday
- No office hour today
  - Makeup office hour tomorrow 11-12 in BWRC
Outline

- Last lecture
  - Latch-based timing
- This lecture
  - Variability and timing
  - Latches

4. Design for performance

B. Statistical timing
Pictorial view of setup and hold tests

- Latest clock arrival time
- Earliest clock arrival time
- Data must be stable
- Setup time
- Hold time
- Early slack
- Late slack
- Actual early AT
- Actual late AT
- Early RAT
- Late RAT
- 0 or more switching(s) allowed

Handling of across-chip variation

- Each gate has a range of delay: [lb, ub]
  - The lower bound is used for early timing
  - The upper bound is used for late timing
- This is called an early/late split
- Static timing obtains bounds on timing slacks
  - Timing is performed as one forward pass and one backward pass

Set up test
- Launching late path
- Capturing early path

Hold test
- Launching early path
- Capturing late path
How is the early/late split computed?

- The best way is to take known effects into account during characterization of library cells
  - History effect, simultaneous switching, pre-charging of internal nodes, etc.
  - This drives separate characterization for early and late; this is the most accurate method
- Failing that, the most common method is derating factors
  - Example: Late delay = library delay * 1.05
    Early delay = library delay * 0.95
- The IBM way of achieving derating is LCD factors (Linear Combination of Delay) (FC=fast chip, SC=slow chip, see next page)
  - Late delay = \( \alpha_L \cdot FC_{\text{delay}} + \beta_L \cdot NOM_{\text{delay}} + \gamma_L \cdot SC_{\text{delay}} \)
  - Early delay = \( \alpha_E \cdot FC_{\text{delay}} + \beta_E \cdot NOM_{\text{delay}} + \gamma_E \cdot SC_{\text{delay}} \)
  - Across-chip variation is therefore assumed to be a fixed proportion of chip-to-chip variation for each cell type

IBM delay modeling*

At a given corner
late delay = intrinsic + systematic + random
early delay = intrinsic – systematic – random

*P. S. Zuchowski, ICCAD’04
Traditional timing corners

The problem with an early/late split

- The early/late split is very useful
  - Allows bounds during delay modeling
  - Any unknown or hard-to-model effect can be swept under the rug of an early/late split
- But, it has problems
  - Additional pessimism (which may be desirable)
  - Unnecessary pessimism (which is never desirable)

This physically common portion can’t be both fast and slow at the same time
Additional pessimism: Clock tree view

How to have less pessimism?

- Common path pessimism removal
- Account for correlations
- Credit for statistical averaging of random
The problem of correlations

- There are many reasons for correlations
  - Chip-to-chip variations are perfectly correlated within a single chip
  - Same circuit types
  - Same device families
  - Same metal levels
  - Same voltage islands
  - Same regions of the chip
  - Dependence on common sources of variation
  - Reconvergent fanout
  - Etc.

- In a reasonable-sized chip, there may be 100 million timing quantities, so we don’t handle correlations in the classical way
  - Not by storing and manipulating a 100M x 100M covariance matrix
### Canonical form

\[ a_0 + a_1 \Delta X_1 + a_2 \Delta X_2 + \cdots + a_n \Delta X_n + a_{n+1} \Delta R_a \]

- All timing quantities are parameterized by the sources of variation
- Correlation can be judged on-demand by inspection

### Statistical timing basics

- Represent all timing quantities in canonical form
  - Delays, slews, guard times, ATs, RATs, slacks, PLL adjusts, constraints, CPPR adjusts
- Propagate ATs forward through the timing graph
  - Addition of two canonical forms is easy
  - Max/min operations are also easy with the help of some analytic formulas
- Propagate RATs backward through the timing graph
  - Subtraction of two canonical forms is easy
  - Use statistical max/min operations
- Slack is simply the difference between AT and RAT
  - Since this is available in canonical form, we get sensitivities of circuit performance to sources of variation for free
  - These can be used to ensure a robust design
### Statistical max operation

\[
A = a_0 + \sum_{i=1}^{n} a_i \Delta X_i + a_{n+1} \Delta N_a
\]

\[
B = b_0 + \sum_{i=1}^{n} b_i \Delta X_i + b_{n+1} \Delta N_b
\]

\[
\sigma_A = \sqrt{\sum_{i=1}^{n+1} a_i^2}
\]

\[
\sigma_B = \sqrt{\sum_{i=1}^{n+1} b_i^2}
\]

\[
\rho = \frac{\sigma_A \sigma_B}{\sigma_A \sigma_B}
\]

\[
\theta = \left( \frac{\sigma_A^2 + \sigma_B^2 - 2\rho \sigma_A \sigma_B}{1} \right)^{1/2}
\]

\[
t = \Phi \left[ \frac{a_0 - b_0}{\theta} \right]
\]

\[
E[\max(A, B)] = a_0 t + b_0 (1 - t) + \theta \phi \left[ \frac{a_0 - b_0}{\theta} \right]
\]

\[
E[\max(A, B)]^2 = (\sigma_A^2 + \sigma_B^2) t + (\sigma_A^2 + \sigma_B^2) (1 - t) + (a_0 + b_0) \theta \phi \left[ \frac{a_0 - b_0}{\theta} \right]
\]

---

### Unified view of correlations

**Independently random part**

**Spatially correlated part:** within-chip distance-related correlation

**Globally correlated part:** chip-to-chip, wafer-to-wafer, batch-to-batch variation

**Distance**

**Correlation Coefficient**

**D = a_0 + \sum a_i \Delta X_i + a_r \Delta X_r**
Spatial correlation vs. early/late split

Dependence on common virtual variables cancels out at the timing test

ICCAD '07 Tutorial
Chandu Visweswarah

4. Design for performance

C. Latches and flip-flops
Latch vs. Flip-Flop

Performance metrics

Delay metrics

- Delay penalty
- Clock skew penalty
- Inclusion of logic
- Inherent race immunity

Power/Energy Metrics

- Power/energy
- PDP, EDP

Design robustness
Latches

Transmission-Gate Latch

C\textsuperscript{2}MOS Latch

Latches

(a) The transparent high latch (THL)  
(b) The transparent low latch (TLL)

(c) Timing waveforms for the THL

Latch Pair as a Flip-Flop

Requirements for the Flip-Flop Design

- High speed of operation:
  - Small Clk-Output delay
  - Small setup time
  - Small hold time → Inherent race immunity
- Low power
- Small clock load
- High driving capability
- Integration of logic into flip-flop
- Multiplexed or clock scan
- Robustness
Sources of Noise

1. Noise on input
2. Leakage
3. α-Particle and cosmic rays
4. Unrelated signal coupling
5. Power supply ripple

Types of Flip-Flops

Latch Pair
(Master-Slave)

Pulse-Triggered Latch
Flip-Flop Delay

- Sum of setup time and Clk-output delay is the true measure of the performance with respect to the system speed
- $T = T_{\text{Clk-Q}} + T_{\text{Logic}} + T_{\text{setup}}$ (ignoring skew)

Delay vs. Setup/Hold Times

Data-Clk [ps] vs. Clk-Output [ps] plot showing minimum data-output delay, setup, and hold times.
Master-Slave Latch Pairs

Case 1: PowerPC 603 (Gerosa, JSSC 12/94)

![Circuit Diagram of Master-Slave Latch Pair]

- Feedback added for static operation
- Locally generated clock
- Poor driving capability

Master-Slave Latches

Case 2: C^2MOS

![Circuit Diagram of Master-Slave Latch Pair]

- Feedback added for static operation
- Locally generated clock
- Poor driving capability
Next Lecture

› Latches and flip-flops