A Clock Tuning Circuit for System on Chip

Ran Ginosar
VLSI Systems Research Center
Electrical Engineering & Computer Science
Technion--Israel Institute of Technology

Yaron Elboim
Oren Semiconductor (now @ intel)

Avinoam Kolodny
Intel, Technion

OVERVIEW

- **IP reuse** in SoC is essential for time to market
- **Speculative physical design** (floorplan and global routing before logic is done) is also essential for time to market
- **Clock distribution** complicates the integration of IP cores in SoC and leads to iterative physical design
- **Programmable clock tuning circuit**: Enhances design flexibility, eliminates design iterations, supports speculative physical design
- Successful in a commercial product
SoC Clock Distribution Network

IP Core or Module
Global Clock Net
Core Internal Clock Net
Core Internal Clock Driver/PLL:
  • Buffer
  • Freq. Multiply
  • Align
External clock
PLL

The Clock Delay Problem (example)
Internal Clock Delay – Source of the Clock Delay Problem

- Internal clock delay \( d \) ≡ Delay from clock input port to all flip-flops inside an IP core / module

The “Min-Delay” Timing Problem

Hold time violation for in2 – The long internal clock delay within IP2 requires a long hold time
Common Solution #1: Data Delay Insertion

- Delay the early signals by inserting delay buffers
- Typically used by synthesizers
- Expensive: area, power 😞

![Diagram of Data Delay Insertion]

Common Solution #2: Clock Delay Insertion

- Align clocks in the SoC
- Turn the SoC into a zero clock skew system
- Hold time violation (example) is removed
- Much less expensive 😊

![Diagram of Clock Delay Insertion]
Two Methods of Clock Distribution Networks

Zero skew at clock inputs to IP cores (a)

Zero skew at the flip-flops (b)

Assume perfectly balanced clock tree!

Clock Delay Design Algorithm

\[ d_i = \text{internal clock delay of IP core } i \]
\[ D = \max\{d_i\} \]
\[ \text{Add delay } D_i = D - d_i \text{ to each IP core } i \]

\[ \downarrow \]

Recommendation: Hard IP core providers should:

- Prefer registered inputs and outputs
- Supply internal clock delay information
Clock Delay Insertion by Global Clock Re-design

- Repetitive process:
  - Speculative physical design
  - When logic changes, physical global nets may need to be changed as well
- Iterations costly in time and engineering resources
- Adverse effect on time to silicon
- Small changes might trigger a complete redesign
- Designers resist changes…

Clock Delay Insertion Using Programmable Clock Delays

Delays are programmed (hard-wired) during final design stages

- No more iterations
- Easier to change logic along the way
- Shorter time to silicon
Side benefit:
Unbalanced clock tree

- Balancing the global clock tree is non-trivial
- Programmable delay lines can balance the tree:

Hierarchical Clock Tuning
Hierarchical Clock Tuning Algorithm

\[ H\text{-tune}(\text{Cell}) \begin{array}{l}
\{ \\
\quad \text{foreach (sub-cell } j \in \text{Cell}) \{ \\
\quad\quad \text{if (sub-cell } j \text{ is a leaf) } d_j = \text{sub-cell } j\text{'s internal clock delay;} \\
\quad\quad\quad \text{else } d_j = H\text{-tune}(\text{sub-cell}); \\
\quad\}\} \\
\quad D = \max\{d_j\}; \\
\quad \text{foreach (sub-cell } j \in \text{Cell}) \text{ add tuning delay } D_j = D - d_j; \\
\quad \text{Return } (D + \text{distribution\_delay(CCell)}); \\
\} \]

The Clock Delay Tuning Circuit

Programmable *digital* “mirror” delay line

Alternative: Tapped delay line with mux’ed outputs
Application in a Commercial SoC

- Saved weeks of design time compared to a previous product
- 0.18μ: Needed only max D = 800 psec
- Inverter delay ~ 45 psec
- Needed only 20 inverters, 10 stages
- Flawless in the product

<table>
<thead>
<tr>
<th>Device Count</th>
<th>Power Dissipation</th>
<th>Process</th>
<th>Package</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 M</td>
<td>&lt;1 W</td>
<td>TSMC 0.18 μm</td>
<td>128 pins QFP</td>
<td>Digital TV Decoder</td>
</tr>
</tbody>
</table>

Die Photo

Global Clock Net

Alternatives

- Tester tuning (laser fuses)
  - Die-specific, may increase yield
- Power-up tuning
  - More complex
- Periodic tuning
  - Compensate for slow drifts (temp, voltage)
- Continuous tuning
  - Zillion PLLs / DLLs
- Asynchronous Interconnect
  - Nothing beats that!
Summary and Conclusions

- Novel clock tuning method:
  - Layout global clock net once
  - Add programmable delays at inputs to every IP core / module
  - Change logic at will
  - Fix clock delays after last extract
- Eliminates design iterations
- Improves design modularity, enables “last minutes changes”
- Fixes unbalanced clock trees
- Successfully implemented in a commercial SoC product. Saved weeks on time-to-market