System-Level Timing Closure
Using IBIS Models

Barry Katz
President/CTO, SiSoft
Asian IBIS Summit
Agenda

• High Speed System Design
• Establishing timing model
  – Derivation of timing equations
  – Idealized timing analysis
  – The role of signal integrity
  – Reconciling signal integrity with timing
• Pre-route exploration
• Driving physical design
• Post-route validation
• Design analysis reuse
• Case study: DDR2 memory
High Speed System Design
... Not Just “Signal Integrity”

- High Speed Design involves multiple disciplines
- Changes in any area drive changes in others
- Mastery of modeling details & process flow is essential for success
System Level Timing Closure

- Successful high speed design requires a rigorous methodology for ensuring positive design margin across all combinations of:
  - Component timing (process)
  - Voltage & temperature
  - Package & PCB routing lengths
  - PCB manufacturing variations ($Z_0$, loss)
Establishing Timing Budgets

- High speed interfaces have one or more “transactions” that require timing closure
- Memory example:
  - Address/control
  - Data read
  - Data write
  - Strobe to Clock
- Timing relationships must be identified and closed for each different transaction
Source-Sync Transaction Example

- Establish component timing & transfer protocol
- Derive timing equations
- Idealized timing analysis
- Signal integrity analysis and Timing Closure
Component Timing, Transfer Protocol

1. Design Goals
   - Clock = 250 MHz
   - Source Sync, DDR transfer
   - Data Unit Interval = 2ns
   - 90° clock shift on PCB

2. Driver Timing
   - CLKOUT
     - [0ns, 0ns]
     - [-0.3ns, 0.3ns]
   - Q0 .. Q15
   - [0ns, 0ns]
   - [-0.3ns, 0.3ns]

3. Interconnect Delays
   - CLKIN
     - [0ns, 0ns]
     - [0.4ns, 0.4ns]
   - D0 .. D15
     - [0.4ns, 0.4ns]
Derive Timing Equations

  = [0ns + a1] – [0.3ns + b2] - [0.4ns]
  = a1 – b2 – 0.7ns

Hold margin = [Data UI] + [early data] – [late clock] - [hold requirement]
  = [2ns] + [-0.3ns + b1] – [0ns + a2] - [0.4ns]
  = 1.3ns + b1 – a2
Idealized Timing Analysis

Minimum data length = 3", at 180ps/in = 0.54ns

CLKOUT
[0ns, 0ns]

Q0 .. Q15
[-0.3ns, 0.3ns]

CLKIN
[0ns, 0ns]

D0 .. D15
[0.4ns, 0.4ns]

Setup margin = a1 – b2 – 0.7ns
= 1.54ns – 0.54ns – 0.7ns
= 0.3 ns

Hold margin = 1.3ns + b1 – a2
= 1.3ns + 0.54ns – 1.54ns
= 0.3ns
The Role of Signal Integrity

- Detailed analysis of digital switching behavior
- IBIS or HSpice models define I/O buffer behavior
- Accounts for
  - Actual circuit loading
  - Reflections / ringing
  - Circuit topology
  - Inter-symbol interference
  - Switching thresholds

Idealized Delays

Real-World Delays
Reconciling SI with Timing

- Static timing and signal integrity measurements must be compatible
- SI measurements are “normalized” to conditions under which loading is specified
  - IBIS Vref, Cref, Rref, Vmeas
- Timing Closure occurs when integrated timing/SI results show acceptable setup/hold margins
Building an Executable Timing Model

- For each interface, all transactions must be validated for all cases:
  - Component timing (process)
  - Voltage, temperature
  - PCB variations
- Creating an executable timing model to perform automatic regression is ideal
- Possibilities
  - Excel
  - Custom scripting
  - EDA tools

\[ t_{cycle} = t_{co} + t_{final \ settling} + t_{setup} + t_{skew} + t_{jitter} + t_{SSO} + t_{ISI} \]
Pre-Route SI Exploration

• Pre-route simulations model planned
  – Drivers
  – Receivers
  – Routing topology & lengths
  – Termination

• Simulated interconnect delays are extracted and plugged back into the Executable Timing Model

• Setup and hold margins are calculated for temperature, process and voltage corners
Driving Physical Design

- Pre-route SI/Timing analysis defines PCB routing rules
- Rules usually include pin ordering, length limits and stub matching
- Driving automated rules into PCB CAD is essential

Match stub lengths to within 0.2”
Post-Route Validation

- Routed topologies are extracted from PCB database and simulated
- Simulated interconnect delays are extracted and plugged back into system timing model
- Setup and hold margins are calculated for temperature, process and voltage corners
Once all the SI/timing data for an interface has been captured, it should be possible to directly reuse that information for multiple instances in a project or other projects.

Each interface kit contains net class schematics, timing data & SI models.
Case Study: DDR2 System Memory

- DDR2 supports one or two DIMM modules
- DIMM Modules
  - Registered and Unbuffered
  - 4 to 18 memory devices
- Two module, data write transaction is presented here
- Complete case study:
  “Features and Implementation of High-Performance 667Mbs and 800Mbs DDRII Memory Systems”
  - Presented by Micron & SiSoft
  - DesignCon West, 2005
DDR2 Data Write Configuration

- Termination strategy is dynamic; depends on how many DIMMs are present and which device is receiving
- Simulation environment must switch receiver models based on which case is being analyzed

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Write to</th>
<th>DQ Active-Term Resistance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Controller</td>
<td>Dram at Slot 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Front Side</td>
</tr>
<tr>
<td>2R / 2R</td>
<td>Slot 1</td>
<td>No Term</td>
</tr>
<tr>
<td></td>
<td>Slot 2</td>
<td>No Term</td>
</tr>
<tr>
<td>2R / 1R</td>
<td>Slot 1</td>
<td>No Term</td>
</tr>
<tr>
<td></td>
<td>Slot 2</td>
<td>No Term</td>
</tr>
<tr>
<td>1R / 2R</td>
<td>Slot 1</td>
<td>No Term</td>
</tr>
<tr>
<td></td>
<td>Slot 2</td>
<td>No Term</td>
</tr>
<tr>
<td>1R / 1R</td>
<td>Slot 1</td>
<td>No Term</td>
</tr>
<tr>
<td></td>
<td>Slot 2</td>
<td>No Term</td>
</tr>
<tr>
<td>2R / Empty</td>
<td>Slot 1</td>
<td>No Term</td>
</tr>
<tr>
<td>Empty / 2R</td>
<td>Slot 2</td>
<td>No Term</td>
</tr>
<tr>
<td>1R / Empty</td>
<td>Slot 1</td>
<td>No Term</td>
</tr>
<tr>
<td>Empty / 1R</td>
<td>Slot 2</td>
<td>No Term</td>
</tr>
</tbody>
</table>
Slew Rate Derating – “Virtual Eye”

Eye at device pad
(simulated result)

Eye at receiver output
(simulated result)

Waveform derating scheme

Virtual eye at receiver
(computed result)

Waveform processing

4.60ns Eye

4.75ns Eye

SiSoft
**DDR2 Analysis Results**

**Data Write Slow / Fast Corners**

<table>
<thead>
<tr>
<th>Setup Margin (ns)</th>
<th>Hold Margin (ns)</th>
<th>Transfer Net</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.336</td>
<td>3.167</td>
<td>addcmd_8L_8L</td>
</tr>
<tr>
<td>No AC specs</td>
<td>No AC specs</td>
<td>ck_4L_slot1</td>
</tr>
<tr>
<td>0.686</td>
<td>0.93</td>
<td>ctrl_4L_slot1</td>
</tr>
<tr>
<td>0.697</td>
<td>0.964</td>
<td>ctrl_4L_slot2</td>
</tr>
<tr>
<td>0.468</td>
<td>0.205</td>
<td>dm_2R_2R</td>
</tr>
<tr>
<td>0.213</td>
<td>-0.148</td>
<td>dq_2R_2R</td>
</tr>
<tr>
<td>1.155</td>
<td>0.944</td>
<td>dqs_2R_2R</td>
</tr>
</tbody>
</table>

---

Asian IBIS Summit - Shanghai, China - October 27, 2006
3.60ns Eye

High-speed system design requires a rigorous, repeatable methodology for achieving **Timing Closure**

- Static Timing, Signal Integrity, and physical design rules are all interrelated

- An Executable Timing Model allows for a user to validate all transactions across all cases

- Signal Integrity analysis must be performed in accordance with the system timing model

<table>
<thead>
<tr>
<th>Setup Margin (ns)</th>
<th>Hold Margin (ns)</th>
<th>Transfer Net</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.336</td>
<td>3.167</td>
<td>addcmd_8L_8L</td>
</tr>
<tr>
<td>No AC specs</td>
<td>No AC specs</td>
<td>ck_4L_slot1</td>
</tr>
<tr>
<td>No AC specs</td>
<td>No AC specs</td>
<td>ck_4L_slot2</td>
</tr>
<tr>
<td>0.686</td>
<td>0.93</td>
<td>ctrl_4L_slot1</td>
</tr>
<tr>
<td>0.697</td>
<td>0.964</td>
<td>ctrl_4L_slot2</td>
</tr>
<tr>
<td>0.468</td>
<td>0.205</td>
<td>dm_2R_2R</td>
</tr>
<tr>
<td>0.213</td>
<td>-0.148</td>
<td>dq_2R_2R</td>
</tr>
<tr>
<td>1.155</td>
<td>0.944</td>
<td>dqs_2R_2R</td>
</tr>
</tbody>
</table>