Comparison of Timing Approaches in VHDL

Victor Berman
Cadence Design Systems
2 Lowell Research Center
Lowell MA 01852
508 934 0276
berman@cadence.com

Agenda

- Current efforts in VHDL Timing
- Differing Goals?
- Developing Requirements
- Topics Under Discussion
- Plans for Progress
VHDL Timing Efforts

- IEEE Timing Working Group
- CFI Timing/Delay Data Constraints Working Group
  - Fitting timing into existing Data Model for Frame Works
  - Working with ASIC Library Representation Working Group
- VI Technical Committee
  - Put out Request for Proposals
- VITAL - industry initiative
  - Looking at re-use of existing timing methodology

No Single Approach Accepted Industry-Wide

Victor Berman

Do These Efforts Have Differing Goals?

- Coming at the problem from different angles
  - IEEE - general solution for VHDL timing
  - CFI - inter-tool communication for Frameworks
  - VI - industry common methodology
  - VITAL - rapid adoption through methodology re-use
- Differ in time frame and emphasis
- Basic Goals are the same

Victor Berman
IEEE Timing Working Group

• Goal - ASIC Library Sign-Off in VHDL
  • Deal with System timing later
• VHDL Orientation
• Issues
  • Data Model for Timing Specification
  • External File vs. All VHDL Representation
  • Modeling Guidelines

Victor Berman

CFI Timing/Delay Data Constraints Working

• Starting Point is 1.0 Data Model
• Language is secondary issue
• Interoperability of Tools Is Primary
  • Same Results from Different Vendors and Tools
  • Completeness of Data Model and Common Procedural Interface is Key
• Agreement on Phased Approach
  • First Define Data Model, Interchange Format, and PI
  • Leave Computational Model for Later Phase

Victor Berman
VI Modeling Technical Working Group

- Looking at Complete VHDL Modeling Issue
- Timing is essential - but other modeling issues addressed
- Group is in early stages
  - Request for Proposals Issued
  - Looking for a Solution That Can Gain Industry Wide Acceptance
  - Needs consensus from membership

VITAL Approach

- Industry Driven
- Method Based on Technology Re-Use
- Clear Goal of Near Term ASIC Library Availability
Developing Requirements

• Requirements Drive the Solution

• Each Group Uses Different Method
  • Generally Ad-Hoc and Dependent on Members of the Group
  • Can be the most difficult and time consuming part of process
  • Important to Have Clear Focus on Goal

• VITAL Decided on Different Approach
  • Survey ASIC Vendors
  • Abstract Requirements From Looking at Their Design Methodology
  • Iterate Findings to Assure Completeness, Correctness

Victor Berman

Developing Requirements

• Summary of ASIC Vendor Survey
  • Most Translate to Internal Format For Final Timing Checks
  • Accuracy of Models is Key Driving Element
  • Maintenance Costs Are More Important Than Development Costs
  • Simulation Speed is A Critical Factor

• Derived Requirements
  • Language is Secondary Issue
  • Single Common Data Format is Key to Accuracy and Maintenance
  • Common Building Block Approach Has Provided High Speed Simulation

Victor Berman
Topics Under Discussion

- Timing Model
- Exchange Format
- Primitive Building Blocks

SDF Abstract Delay Models

$Dev = \text{DEVICE}$ delays (Intrinsic delays)

$lorP = \text{INTERCONNECT}$ delays or PORT delays

$IO = \text{IOPATH}$ delays (Cell path delays)
Key Delay Types

- **I/O Path Delays**
  - Represents the delay between an input and an output
  - May be dependent on a specific type of "edge"
    - (Instance x.y.z)
    - (Delay (IOPATH (posedge i1) o1 < delay_spec>))
    - (IOPATH i2 o1 <delay_spec>))

- **Device Delays**
  - Associates a delay with an output port
  - It may be the same for all output ports on a device or specific to the port
    - (Instance x.a.b)
    - (Delay (DEVICE o1 < delay_spec>))

---

Key Delay Types

- **Interconnect Delays**
  - Represent wire path delays, estimated or actual
    - (Instance x)
    - (Delay (INTERCONNECT y.z.o1 w.i3< delay_spec>))

- **Port Delays**
  - Allows association of a delay directly with an IN port
  - Does not require an IN/OUT pair, thus generalizing the concept
    - (Instance x)
    - (Delay (PORT a.b.il < delay_spec>))
Comments on SDF

- Delay type shown are examples, others are defined and user defined delays can be supported
- Definition of CELLTYPES as well as specific instances provides flexibility
- Timing specifications are general, example shows min, typ, max - can also include conditional or edge sensitive delays
- Header Sections provides Configuration Management data
- Timing checks and constraints are also supported

SDF Current Limitations

- State Transitions Currently Based on ‘0’, ‘1’, ‘Z’
  - Most be generalized for “MVL-9”
  - Handling of ‘X’ transitions most be defined
  - Edge sensitivity most be generalized
- Naming And Syntax Issues Most Be Resolved
  - VHDL-92 Is Defining Syntax for Path_Names and Identifiers
  - Exchange Format Should Follow These New Rules
  - Extended Identifier Syntax Will Allow Traditional Part Names
SDF Information Model

- Work Being Done At University of Manchester
  - Zahir Moosa and Hilary Kahn - CAD Group
- Draft Currently Being Reviewed
- Will Help in Formalizing Specification
  - Defining Well Formed Semantic Model
  - Investigating Completeness and Consistency

Future Plans

- Coordinate Industry/Standards Groups
  - Maximize Synergy
  - Keep Sight of Positive Goals
- Complete ASIC Vendor Requirements Document
- Develop Proposal Based on Requirements
  - Modeling Guidelines
  - Standard Components - gates, cells
  - Timing Models
Conclusions

• Need for Improved Timing Methodology is Critical to VHDL Success
  • VITAL Initiative Has Sparked Wide Spread Interest
  • Several Industry Groups Working the Problem
  • Need Coordination for Different Perspectives

• SDF Format Useful
  • Flexible Representation of Common Timing Constructs
  • Promotes inter-tool communication
  • Needs Refinement for VHDL-92

• Common Industry Goal Should Lead to Success