Synopsys, Inc.
700 East Middlefield Road
Mountain View, CA 94043 USA

CCS Timing

Technical White Paper

Version 2.0

Abstract

This document describes the Synopsys CCS Timing model for accurate and efficient cell-level delay calculation.

12/20/06

Synopsys, Inc.
700 East Middlefield Road
Mountain View, CA 94043 USA

Copyright Notice and Proprietary Information

1 Introduction

Accurate delay calculation is critical for timing closure of complex digital designs. At 90nm and below, physical effects and design styles present new challenges for delay calculation. Top-level interconnect is becoming more resistive with narrower metal widths, resulting in cases where the interconnect impedance is much greater than the drive resistance of the driving cell. Analysis is needed across a wide range of V_{dd} values to support dynamic IR drop effects, and low-power design styles including voltage islands and dynamic voltage/frequency scaling. Inverted temperature dependence at low voltages requires analysis at intermediate temperature values.

A delay model is needed that enables accuracy close to circuit simulation, but with fast calculation to support flat analysis of the largest designs. The model must support calculation of cell delay, interconnect delay, pin slew (also called “transition time”) and input pin capacitance for stages including detailed parasitics. This paper describes Synopsys’ Composite Current Source model for timing (CCS Timing) which addresses these needs and future delay calculation needs.

Delay calculation is performed for one stage at a time, where a stage consists of the driving cell arc, the output RC network, and the capacitance of the network load pins. The goal is to compute the response at the driver output and at the network load pins, given an input slew or waveform at the driver input, as shown in Figure 1. The computed responses are then used to determine the cell delay for the driver and the input slews at the load pins.

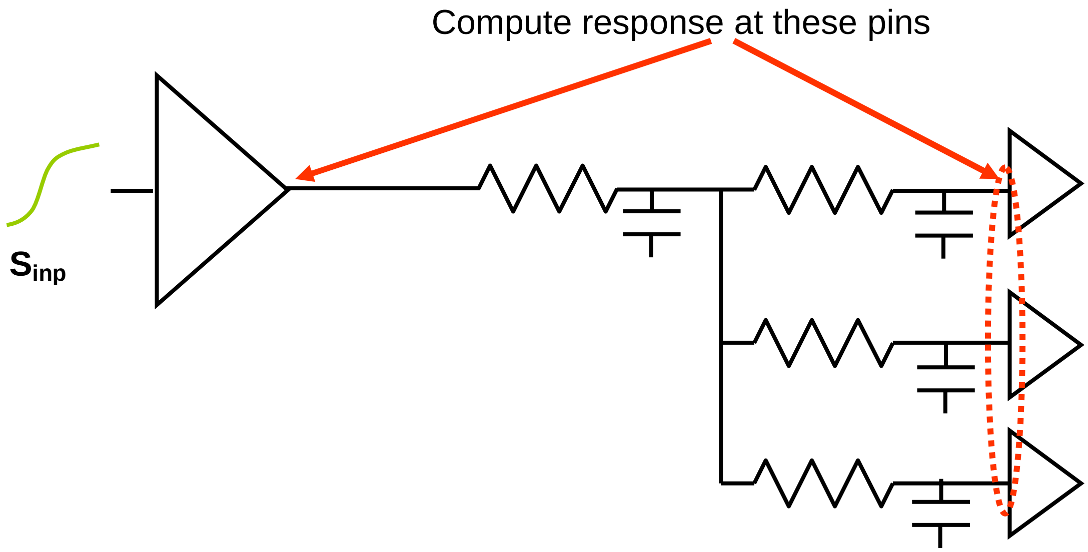

Figure 1: Stage Delay Calculation

To perform stage delay calculation efficiently, three models are created: the driving cell arc is replaced by a driver model; the interconnect RC network is replaced by a reduced order model, such as Block Arnoldi; and the load pins are replaced by a receiver model (see Figure 2).

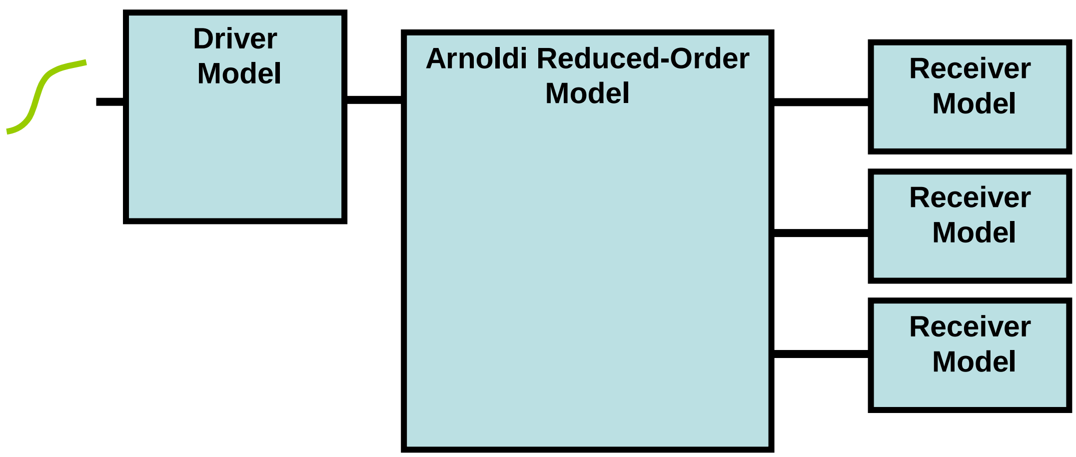

Figure 2: Stage represented by driver model, ROM and receiver models

Note that the receiver model must represent the complex input capacitance of a cell input pin. The transistors do not present a constant input capacitance to a driver. The equivalent capacitive load (from I = C * dv/dt) can vary depending on the rise/fall direction of the transition, the input slew at the pin, the output load, and the state of the cell. In addition, this capacitance can change during the transition. The receiver model must be able to represent all these effects.

2 Previous Approaches

2.1 Thevenin and Norton Models

Previous driver models used either a time-dependent voltage source in series with a resistor (Thevenin model) or a time-dependent current source in parallel with a resistor (Norton model). The resistor in those models is typically referred to as the "drive resistor" and is used to express the timing arc's sensitivity to output capacitance, whereas the waveform shape itself is primarily expressed by the voltage or current source.

Refinements to these models to account for complex aspects of transistor behavior have typically dealt with making the time-dependent nature of the voltage/current source more complex. Other approaches have dealt with multiple drive resistances and arbitrary dynamic impedances.

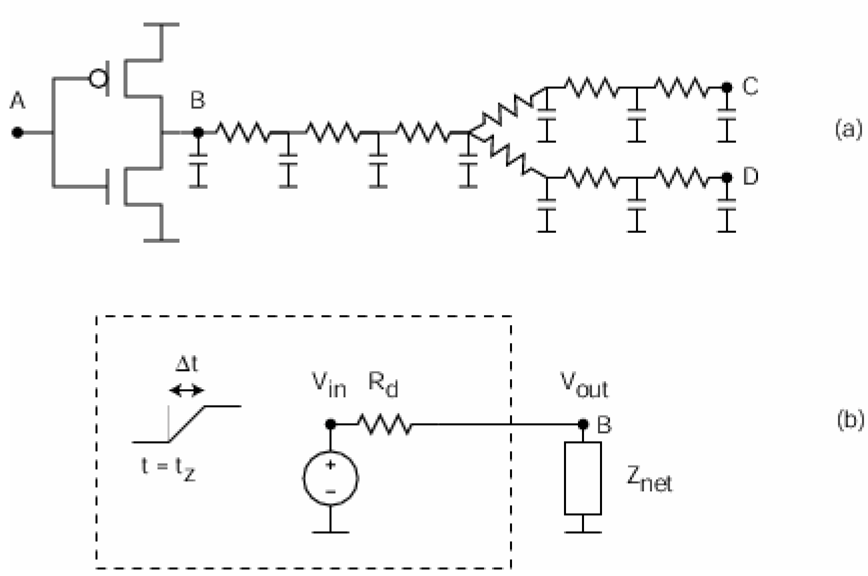

Figure 3: The Rd << Znet problem. (a) shows a transistor circuit driving a detailed parasitic network at node 'B'. (b) The network presents an impedance Znet to the Thevenin driver model. When Rd << Znet, Vout approaches Vin and the driver model can lose accuracy.

Unfortunately, there is a major limitation when conventional models are used to drive an interconnect network with an impedance Z_{net} much greater than the drive resistance R_d. Consider the Thevenin model driving a detailed parasitic network as shown in Figure 3. Note that the circuit forms a voltage divider with

$$\frac{V_{out}}{V_{in}} = \frac{Z_{net}}{R_d + Z_{net}}$$

which approaches unity when $R_d << Z_{net}$. This indicates that a driver model based upon a drive resistance (or arbitrary impedance) that is set independent of the network load will be ineffective in this regime.

Since the transistor behavior deviates from the Thevenin voltage source nearest the power rails, this situation is typically worst when the network delay is greater than the output transition time.

2.2 Previous Receiver Models

The traditional receiver model is a single value of capacitance for an input pin. More recently, separate values have been allowed for rising vs. falling transitions, and a min/max range has been introduced that can bind the complex capacitance effects, but which leads to pessimism during analysis.

Using a single capacitance value for the entire transition results in inaccuracy for single-stage cells where the Miller effect is significant, affecting the calculation of both cell delay and slew. Figure 4 shows that the voltage waveform for the input of a single-stage cell, such as an inverter, cannot be approximated well by any single capacitance value.

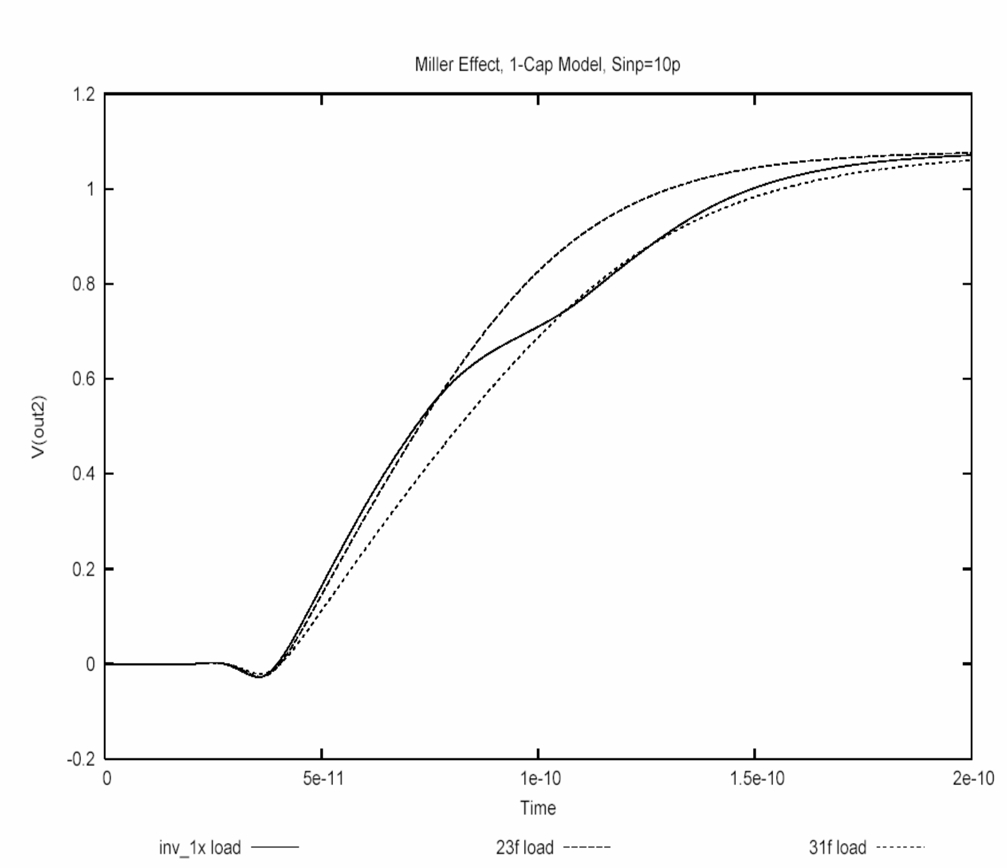

Figure 4: A single capacitance value is insufficient when Miller effect is large

3 The CCS Timing Solution

CCS Timing consists of a driver model and a receiver model. The driver model describes how a timing arc propagates a transition from input to output, and how it can drive arbitrary RC networks. The receiver model describes the capacitance that an input pin presents to driving cells.

The CCS Timing driver model is a time and voltage dependent current source with an essentially infinite drive-resistance, which provides high accuracy even when R_d is much less than Z_{net}. The model achieves this accuracy not by modeling the transistor behavior, but by mapping the arbitrary transistor behavior for lumped loads to the behavior for an arbitrary detailed parasitic network. The following figure shows how the mapping algorithm basically works.

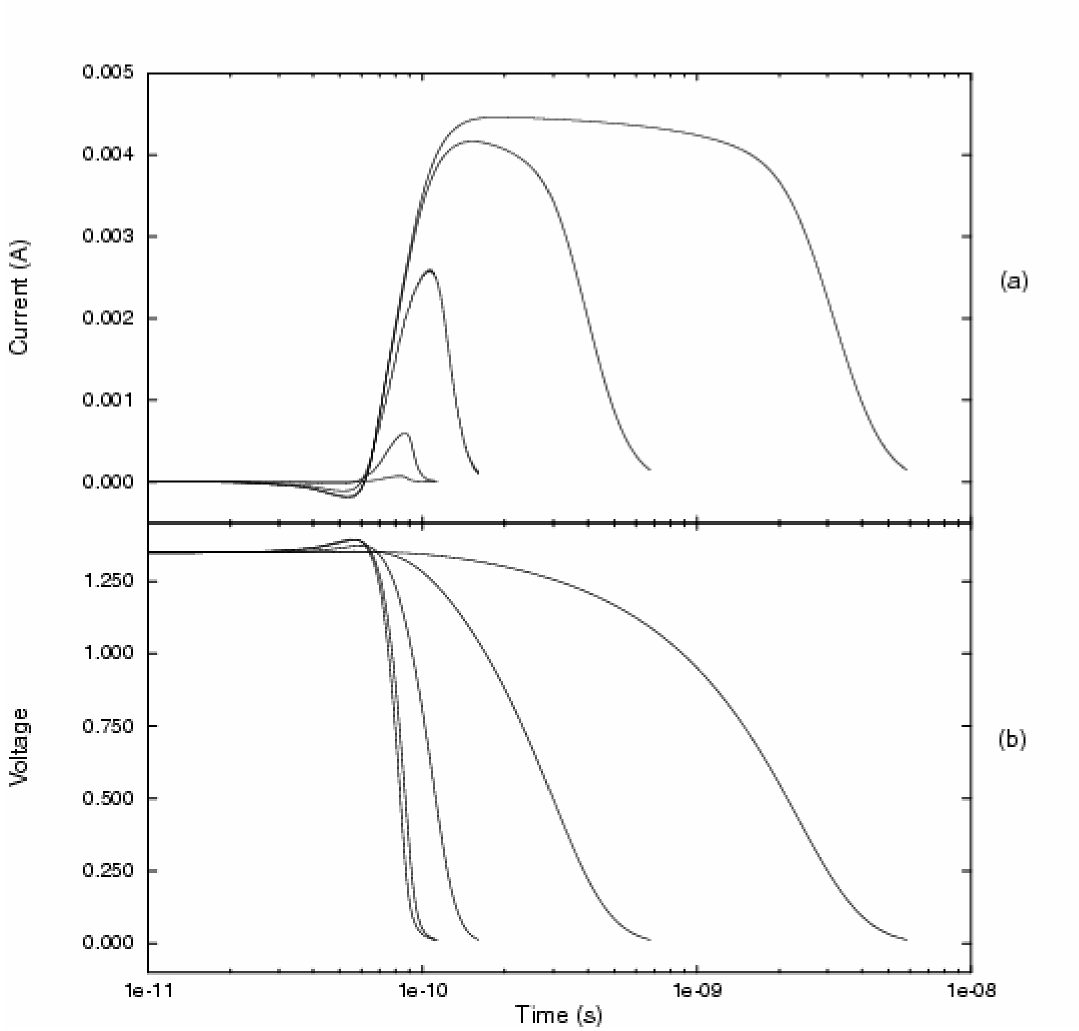

Figure 5: Output current and voltage responses for a falling timing arc. Transistor-level simulation results are shown for different values of load capacitance (1fF, 10fF, 100fF, 1pF, 10pF). (a) Inverted current responses. (b) Voltage responses.

Consider a set of pre-characterization measurements of the output current as a function of time for a specific input slew and a set of output capacitances (Figure 5). When we apply these currents to their respective capacitances, we can reconstruct the voltage waveforms. If we are presented with an output capacitance that we did not pre-characterize with, we can interpolate between the currents to predict the resulting waveform. Similarly, if we are presented with an input slew that we did not use for pre-characterizing, we can also interpolate.

Now consider driving a detailed parasitic network. At a given timestep we can apply the output currents from our pre-characterization to the network. There will be a unique current that will elicit the same voltage on both a lumped capacitance and the network at the given timestep. This current is the chosen value for the given timestep, and we reapply this procedure at every subsequent timestep. In other words, we are applying $I_{out}(V_{out})(t)$.

CCS Timing delay calculation uses advanced interpolation technology to determine a current waveform when the input slew and/or output load values do not match those used during cell characterization. Additionally, interpolation is used for intermediate values of V_{DD} and temperature by using data from multiple libraries.

4 Characterization for CCS Timing

Characterizing a cell timing arc for CCS Timing is very similar to characterization for nonlinear delay models (NLDM): an input stimulus is chosen to produce a specific input slew time (S_{inp}) ; a load capacitance (C_{out}) is connected to the output pin; and a circuit simulation is run in the same way as for NLDM. But instead of measuring voltage thresholds at the output pin, current is measured through the load capacitor and into the input pin. The current through C_{out} is used for the driver model, and the current into the input pin is used to determine the receiver model.

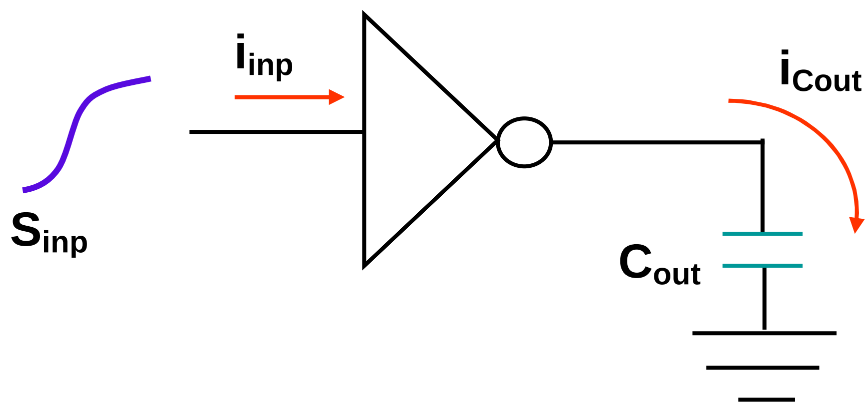

Figure 6: CCS Timing characterization measurements

These characterization experiments are repeated for a table of different S_{inp} and C_{out} combinations. The current through C_{out} is saved for every circuit simulation timestep and then reduced to a much smaller set of current and time (i, t) points. The points are chosen such that $V_{out}(t)$ can be accurately reproduced for every timestep during the transition. Figure

7 shows an example of the complete i(t) waveform and a reduced set of points.

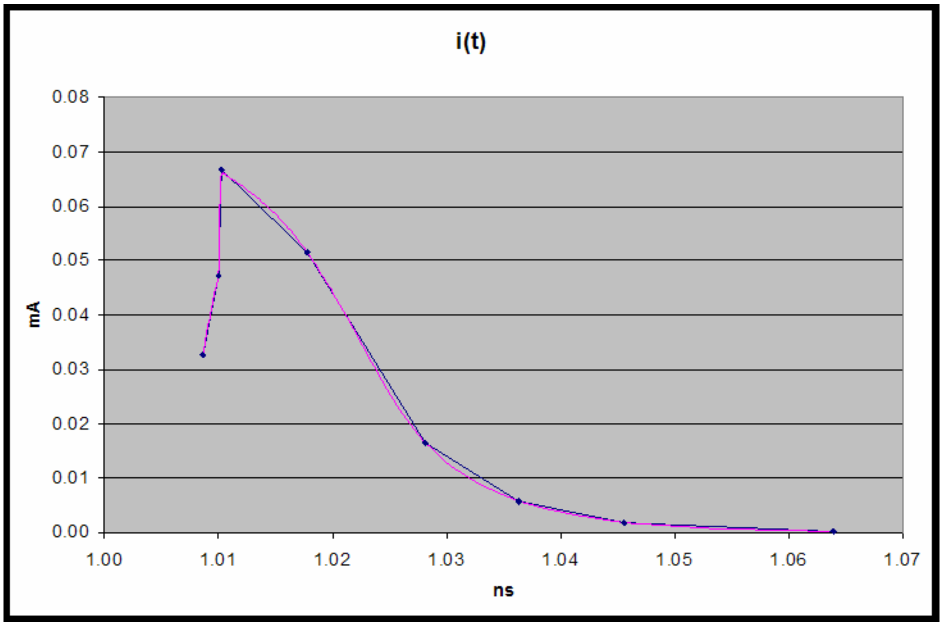

Figure 7: Current waveform from circuit simulation, and reduced current points

The current and voltage at the input pin are saved and then used to determine C1 and C2 values such that gate-level delay calculation can closely match times to the delay threshold and to the second slew threshold at the input pin.

An additional piece of information, input reference time, is needed in order to calculate cell delays. The reference time is the simulation time at which the waveform at the input pin crosses the rising or falling delay threshold (often this is 50% of V_{DD}).

More details about CCS Timing characterization can be found in the Synopsys document titled “Characterization Guidelines for CCS Timing.”

5 Benefits of CCS Timing

CCS Timing delay calculation provides a high accuracy response for cell delay, interconnect delay, and pin slew. Figure 8 shows an example of the CCS Timing response compared with HSPICE on a highly-resistive net.

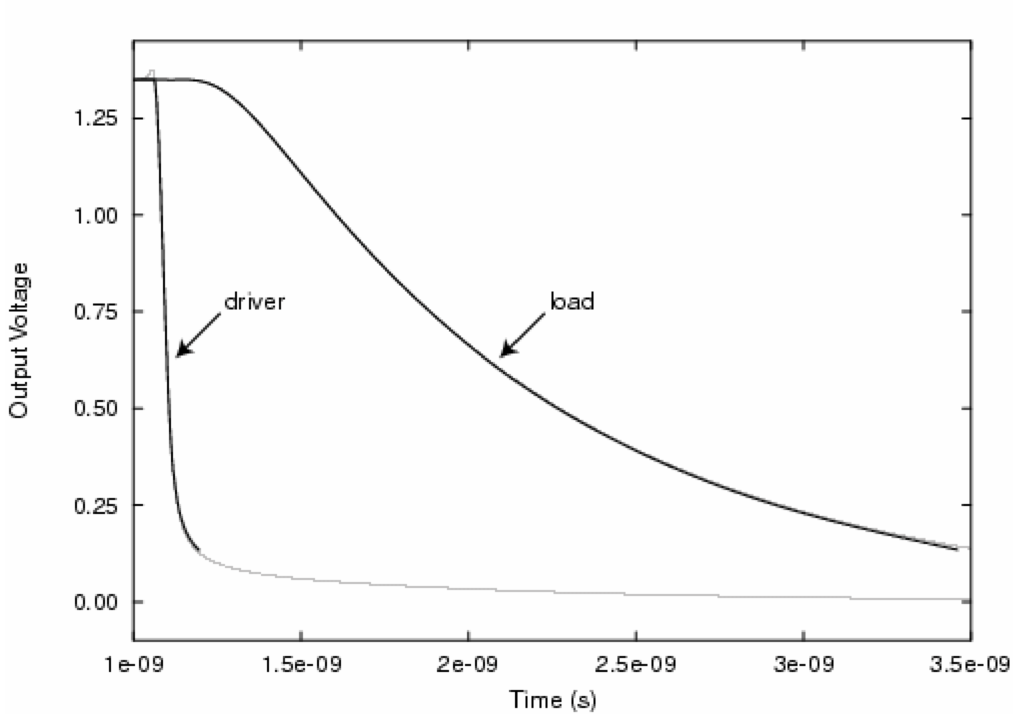

Figure 8: CCS Timing results on stage of a high-drive cell and high-impedance net. Error is less than 1ps.

The CCS Timing receiver model produces excellent results on single-stage cells with large Miller effect. Figure 9 shows how the two-capacitance model allows CCS Timing to match cell delay and transition time with high accuracy.

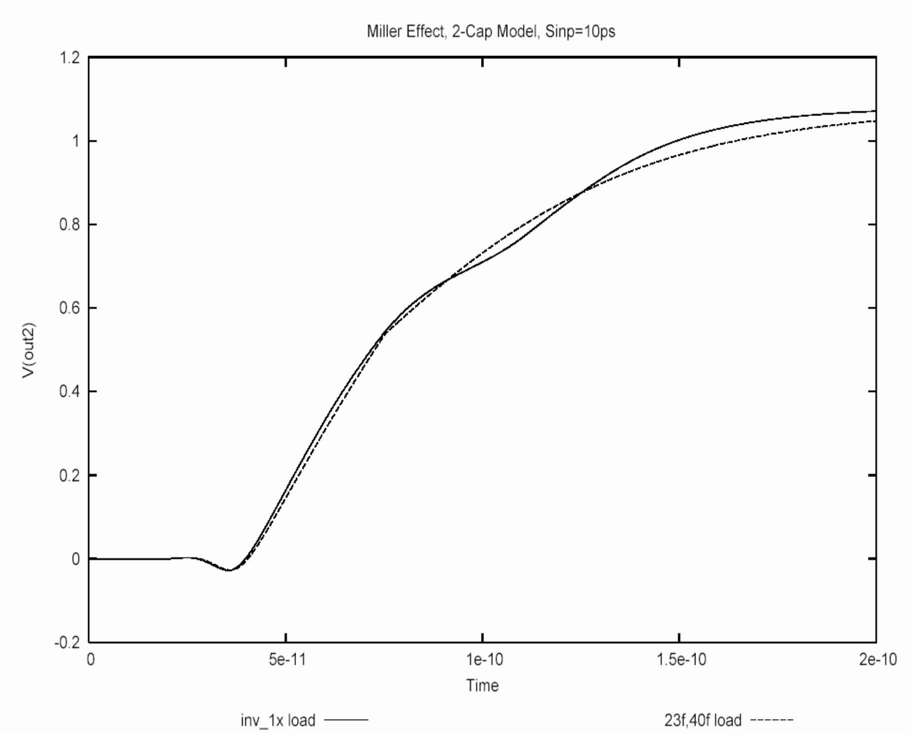

Figure 9: CCS Timing receiver model matches both delay and slew. This shows the voltage waveform at the input of an inverter with large Miller effect. The dashed line is the response with CCS Timing; solid line is from HSPICE.

CCS Timing stage delay and slew results are typically within 2% of the golden circuit simulation values. Figure 10 shows a comparison of CCS Timing versus HSPICE for a large number of test cases, including highly-resistive nets.

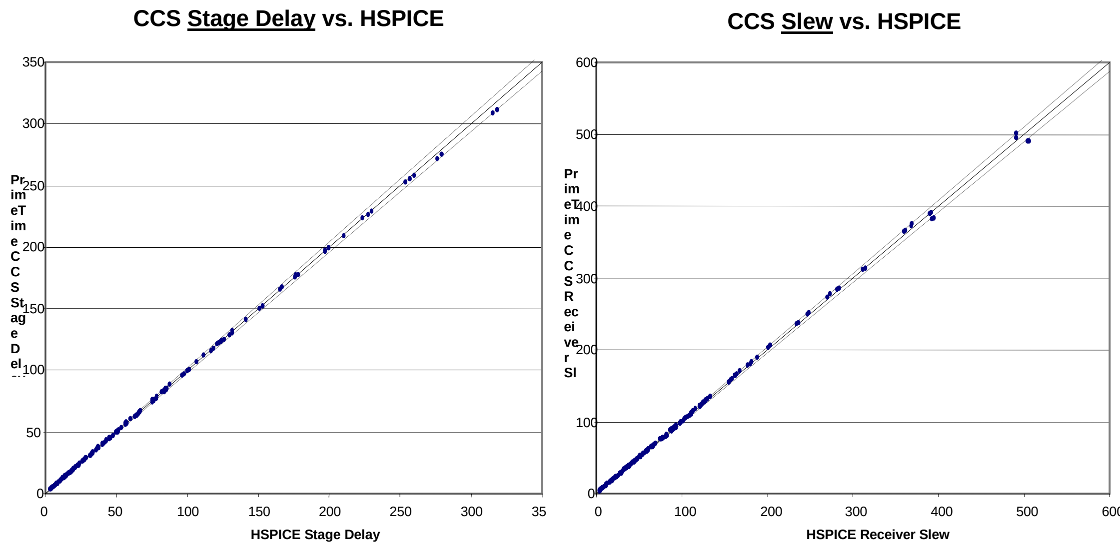

Figure 10: CCS Timing results vs. HSPICE

CCS Timing enables scaling for intermediate V_{DD} and temperature values. Library characterization is done for a small number of V_{dd} values, with advanced current waveform interpolation at runtime. Calculation can be done for any instance-specific value in a continuous range of V_{dd}. This is a key element of flows considering the timing effect of IR drop, and also supports multi-V_{dd} and DVFS (Dynamic Voltage and Frequency Scaling) designs. CCS Timing scaling also supports delay calculation for arbitrary temperature values between characterization points.

Driver model and receiver model data are both scaled. In addition, timing check arcs such as setup, hold, recovery and removal are also scaled. Some timing checks exhibit nonlinear dependence on V_{dd}, as shown in Figure 11.

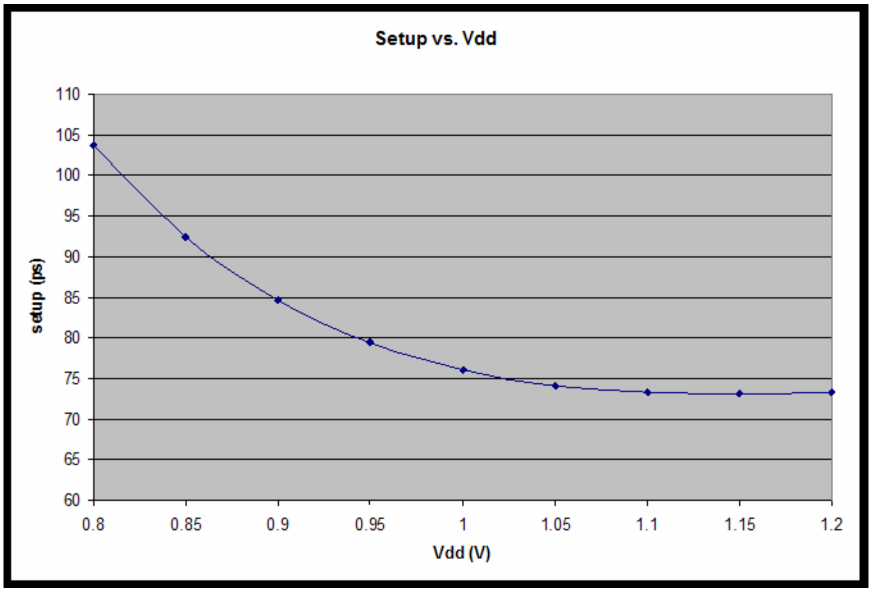

Figure 11: Setup time dependence on Vdd

Synopsys delay calculation with CCS Timing includes powerful nonlinear V_{dd} scaling for timing check arcs. This results in better correlation to circuit simulation than with simple linear interpolation approaches.

The current waveforms are expected to consume larger space in terms of data size compared to the NLDM models. Therefore, a “Compact CCS” is used to represent the current waveforms in a very compact form. The compact CCS takes advantage of similarity of I/V curves in the library. The compact CCS modeling uses a common set of I/V curves (known as base-curves) for the entire library and each instantiation of the current waveform is derived from one of these base curves. This technique allows for high accuracy while reducing the library size by up to 3 to 4x compared to the expanded (non-compact) CCS timing library. For further details please refer to the “Compact CCS Format” specification[1].

CCS timing also allows for accurate representation of current characteristics of the library subjected to the process variation. The variation-aware extension of CCS timing captures the current waveforms as the cell is subjected to process variation with respect to the process parameters. Despite this additional information, the variation-aware extension of CCS timing library size remains reasonably small by using the CCS compact format. For further details please refer to the “CCS Variation Aware Extension” document[2].

6 Summary

CCS Timing is an enabling technology for high-accuracy delay calculation at 90nm and below. Stage delay accuracy is typically within 2% of golden circuit simulation results. The powerful receiver model captures dependence on input slew, output load, direction of switching, and cell state. The two-capacitance approach enables a dynamic calculation that closely matches circuit simulation for inputs susceptible to Miller effect. The CCS driver model can be applied to arbitrary interconnect networks, and produces excellent results on difficult high-impedance nets.

CCS Timing supports the most advanced design styles and flows, including multi-V_{dd} and DVFS designs with accurate scaling for V_{dd} and temperature.

7 References

1. Compact CCS Format specification, http://www.synopsys.com/products/libertyccs/libertyccs.html.
2. CCS Variation-Aware Extension, http://www.synopsys.com/products/libertyccs/libertyccs.html.