Accurate Timing and Noise Analysis of Combinational and Sequential Logic Cells Using Current Source Modeling

Shahin Nazarian[1], Hanif Fatemi[2], Massoud Pedram[3]

[1] Magma Design Automation, Inc., San Jose, CA
[2] Synopsys Inc., Mountain View, CA
[3] University of Southern California, Los Angeles, CA

***Abstract* - A current source (CS) model for CMOS logic cells is presented which can be used for accurate noise and delay analysis in CMOS VLSI circuits. CS modeling is broadly considered as the method of choice for modern static timing and noise analysis tools. Unfortunately, the existing CS models are only applicable to combinational logic cells. In addition to multi-stage logic nature of the sequential cells, the main difficulty in developing a CS model for these cells is the presence of feedback loops. This paper begins by presenting a highly accurate CS model for combinational logic cells, followed by models for common sequential cells including latches and master-slave flip-flops. The proposed model addresses these problems by characterizing the cell with suitable nonlinear current sources and capacitive components. Given the input and clock voltage waveforms of arbitrary shapes, our new model can accurately compute the output voltage waveform of the sequential cell. Experimental results demonstrate close-to-SPICE waveforms with three orders of magnitude speedup.[1]**

I. INTRODUCTION

The drastic down scaling of layout geometries to 65nm and below has resulted in a significant increase in the packing density and the operational frequency of VLSI circuits. An unfortunate side effect of this technology advancement has been the aggravation of noise effects, such as the capacitive crosstalk noise. The conventional static timing analysis (STA) techniques model signal transitions as saturated ramps with known arrival and transition times and propagate these timing parameters from the circuit primary inputs to the primary outputs. To check whether the circuit meets the timing goals, the required time for each circuit node is calculated by using a backward propagation method [2]. If the signal arrival time is less than its required time, the node will be safe from a timing point of view. This signal model has also been used in statistical static timing analysis (SSTA), where the mean and variance of the arrival/transition times are calculated and propagated through the circuit for the purpose of timing analysis. Note that different waveforms with identical arrival time and slew (transition) time applied to the input of a logic gate or an interconnect line can result in very different propagation delays through the component depending on the exact form of the applied signal waveform [1]. Therefore the shape of the voltage waveforms should be considered in order to ensure accurate timing and noise analysis results in sub-90nm CMOS designs.

In the ASIC design flow, combinational and sequential logic cells are pre-characterized for the input-to-output propagation delay and output slew as a function of the input slew and effective output capacitance (C_{eff}). This characterization is based on an implicit assumption about the saturated ramp form of the voltage waveforms that drive the inputs of a logic cell or are produced at its output. We shall refer to this modeling technique as the voltage-based method throughout this paper. The C_{eff} approximation can result in high amount of timing inaccuracy especially in the presence of coupled interconnect. Moreover, voltage-based approach is inherently incompatible with the arbitrary shapes of voltage waveforms, and thus, falls short when dealing with noisy inputs such as crosstalk-induced noisy waveforms. A current source (CS) model is load independent and can handle any electrical waveform at intermediate signal lines of the circuit;

[1] This paper is a major extension of the combinational CSM model introduced in our conference paper published in the Proceedings of the 2006 Design Automation Conference.

therefore, it overcomes the above-mentioned shortcomings of the voltage-based models.

The incompatibility of the voltage-based pre-characterization data with noisy waveforms necessitates additional waveform-aware characterization steps of the logic cells for the purpose of noise analysis. One aspect of the noise analysis is to realize whether a certain noise glitch causes a failure, meaning it is large enough to change the state of a memory element and result in functional error. To perform noise analysis, first the victim noise glitch injected by the aggressor net should be calculated. A mechanism based on noise failure criteria should then be used to determine whether the noise is faulty. Noise failure criteria has been commonly modeled as either DC or AC transfer curves of the receiver logic cell to represent how much a cell is immune to noise glitch [3]. An example is the typical NIC (Noise Immune Curve) which has a hyperbola shape as a function of noise height and width.

Accurate determination of noise failure criteria for sequential elements is very challenging because the final state of the memory element depends not only on input noise height and width but also on its alignment with the clock edge. In [4] noise analysis is performed for feedback loops to check whether the noise transferred from the output back to the input, is strong enough to change the state of the circuit. Considering the fairly complex architecture of sequential cells, especially the presence of feedback loops typical noise analysis pre-characterization is computationally very expensive. A key advantage of our CS model is that it can handle any type of input voltage waveform including full-swing hazardous pulses and partial glitches e.g., a crosstalk-induced noise glitch. Consequently, no extra characterization steps, such as the one in [4], are needed.

Before going into the existing CS models, the two well known vendor formats namely ECSM (Effective Current Source Model) [5] and CCSM (Composite Current Source Model) [6] are briefly reviewed. For a given input slew and C_{eff} , ECSM stores the times at which the output voltage waveform crosses certain pre-defined α% threshold points. In CCSM the output current values at specified voltage level points are stored. It is interesting to note that the stored current values in CCSM can be retrieved using ECSM stored voltage values, and vice versa (from $i_o(t) = C_{eff}\, dv_o/dt$), therefore, ECSM and CCSM are essentially identical models. Despite their names, ECSM and CCSM do not really use a current source model, because the stored data for both vendor formats is a function of the input slew and effective output capacitance C_{eff}, rather than the input and output voltages. In fact, both models can be regarded as generalizations of the conventional cell delay models which only store three pre-defined α% voltage crossing points (such as 20%, 50% and 80%) in the form of cell delay and output slew time as a function of input slew time and C_{eff}. Because CCSM and ECSM store more than three points, they are more accurate than conventional STA tools as long as the input voltage waveform is not noisy. ECSM and CCSM come short in the presence of noisy waveform. This is why the EDA vendors have come up with other models and formats for the noise analysis in VLSI interconnect.

The authors of [7] were among the first to present a true CS model (CSM) of a CMOS logic cell (called Blade) in which a pre-characterized current source is utilized to capture the non-linear behavior of the cell with respect to the input and output voltage values. They model parasitics of the logic cell with a single capacitance at output node. The computed output voltage waveform is time shifted by a pre-characterized value to compensate for the offset with respect to Hspice. The Miller effect between the input and output nodes was ignored in this model. In [8], a Blade-based model is used and the input and output voltage waveforms are approximated with Weibull functions. Keller et al. [9] presented a CS model for the purpose of crosstalk noise analysis. Similar to Blade, a pre-characterized current source is used. The parasitic components, namely the Miller and the output capacitances are assumed to be constant regardless of the input and output voltage values. In practice, these capacitive effects can vary by orders of magnitude depending on cell input and output voltage values. In [10] this weakness is resolved by introducing a nonlinear output capacitance model. In [11] the authors presented a CSM in which the input and output pins as well as several chosen internal pins of the cell are modeled with a voltage-dependent current source and a nonlinear capacitor. Each component in this model generally depends on all the input voltage and the output voltages. In [12] we introduced nonlinear input, output, and Miller capacitors along with an output current source, all of which are functions of the input and output voltages.

In addition to being independent of the C_{eff} and capable of handling any arbitrary shape waveform, CSM is compelling in the sense that instead of only propagating the delay and slew value, it can propagate the whole voltage waveform (in the form of a set of <time, voltage> pairs). CSM is able to do this propagation along the whole timing path from primary input to primary output. High accuracy of the CS models makes them attractive for employment inside a signoff timing analysis tool. Once a set of critical paths is identified by a standard static timing analysis tool, CS models of logic cells along a target critical path may be utilized to provide an accurate, yet highly efficient, evaluation of the timing criticality and/or noise susceptibility of the path in question. Close-to-SPICE accuracy with orders of magnitude faster than SPICE tools, make the CSM-based analysis very attractive. For example in [9] an efficient CS-based technique for worst case aggressor alignment is described that can reduce the pessimism of the conventional voltage based techniques by 50%.

All previous CSM approaches have targeted combinational logic cells. However, each combinational part of the circuit is fed and the output results are captured by a set of sequential cells. Therefore, lack of CSM for sequential circuit elements makes it impossible to have a complete CSM-based solution for performing the delay and noise analysis and optimization steps. Our CSM for the sequential cells makes it possible to construct the exact voltage waveforms for their outputs, and hence, drastically reduce the pessimism of timing arc calculations and setup/hold tests. To the best of our knowledge, we are the first to introduce current source modeling of the sequential cells.

One of the deficiencies of typical sequential cell models is that they report an unknown result for the output if the setup/hold time tests are violated. A key benefit of the proposed model for CMOS register cells is that the output waveform may be computed even when setup/hold time violations occur. This can be very useful for diagnostic purposes.

The major contributions of our work are:

- A more accurate current source model for combinational cells is presented.
- Current source modeling is introduced for sequential cells, e.g., latches and flip-flops.
- A thorough investigation is conducted for different circuit elements in sequential cells and their effects on the output voltage waveform calculation process. The feedback is the most important and challenging element.
- The cell output voltage waveform can be constructed with close-to-SPICE accuracy (average mean squared error with respect to SPICE for voltage waveforms was less than 2% of V_{dd}) with speedup as high as 2000 times compared to Hspice [13]. This is achieved because the cell parasitic effects such as the Miller capacitance, the nonlinearity of these parasitic effects, and the feedback and multi-stage loading effects of sub-circuits are captured by our pre-characterized CSM.
- The output of the cell can be predicted even when timing tests are violated. Voltage-base sequential models report "unknown" for the output if a timing constraint such as setup check is not met.

The remainder of this paper is organized as follows. In section II Our CS model for combinational logic cells are presented. Our sequential CS model is presented section III in two subsections of pre-characterization and output voltage calculation. Sections IV and V explain the CS model for Master-Slave flip-flops and SR latches respectively. Finally, experimental results and conclusions are presented.

II. CS Modeling – Combinational Logic Cells

We first start with our CSM model for combinational logic cells. This will lead us to a better understanding of our proposed model for sequential circuit elements.

Various CSMs for combinational logic cells are essentially similar in the sense that they all model the output current of the logic cell with a voltage-dependent current source. A DC analysis step is performed to pre-characterize this current source as a function of the input and output voltages of the cell. The difference between the existing combinational CSMs is mainly on how they capture the parasitic effects. The main motivation for us to create a new CS model was that the existing models sometimes exhibit rather large errors compared to Hspice, because they ignore parasitic effects altogether or make simplistic assumptions about them.

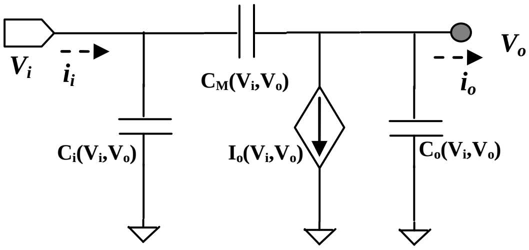


Figure 1. CSM for a combinational logic cell [12].

Our combinational CSM model which is shown in Figure 1 consists of three nonlinear voltage-dependent capacitive components, namely, input and output parasitic capacitances, $C_i(V_i,V_o)$ and $C_o(V_i,V_o)$, to model the parasitic effects at input and output nodes of the cell and the Miller capacitance, $C_M(V_i,V_o)$, to model the Miller effect between the two nodes. The model also has $I_o(V_i,V_o)$, a nonlinear voltage-controlled current source at the output node. Each component is a function of the input and output voltage values. The following subsections give details of our pre-characterization steps for the CSM of Figure 1 using Hspice.

The current source at the output node captures the non-linear resistive behavior of the combinational logic cell during an output transition. More precisely, the following KCL equation models the current at the output pin of the cell during switching:

$$\begin{aligned} &i_o + I_o(V_i,V_o) + \left(C_o(V_i,V_o) + C_M(V_i,V_o)\right)\frac{dV_o}{dt} \\ &\quad - C_M(V_i,V_o)\frac{dV_i}{dt} = 0 \end{aligned} \qquad (1)$$

A. Pre-*characterization*

Model parameters, $I_o(V_i,V_o)$, $C_o(V_i,V_o)$, $C_M(V_i,V_o)$ and $C_i(V_i,V_o)$ should be calculated and stored in the logic cell delay library. This library can then be imported and used by the timing analysis tool during logic cell timing calculations. The process of calculating and storing the required model parameters is called *characterization*. Since logic cells in the library are typically characterized once before doing any timing analysis, the process is sometimes referred to as *pre-characterization*.

To characterize $I_o(V_i,V_o)$, input and output pins are driven by DC values. Each pin is swept from $(-\Delta)$ to $(V_{DD}+\Delta)$ where Δ is considered for cases where the input and output voltages under/over shoot beyond ground and V_{DD}. The current sourced by the output pin is measured in SPICE. As a result, a 2-D lookup table is constructed to store the values of I_o.

Figure 2 shows the characterization setup for calculating the model elements which is then stored in the cell library. To characterize $I_o(V_i,V_o)$ for the cell, CH1 and CH2 are DC voltage sources which are swept from $(-\Delta)$ to $(V_{DD}+\Delta)$. Since V_o and V_i do not change, all derivative terms in Equation (1) become zero i.e.,

$$i_o + I_o(V_i,V_o) = 0 \qquad (2)$$

For given input-output voltage pair V_{CH1} and V_{CH2}, it is enough to monitor i_o, the current flowing into CH2, to determine $I_o(V_{CH1},V_{CH2})$.

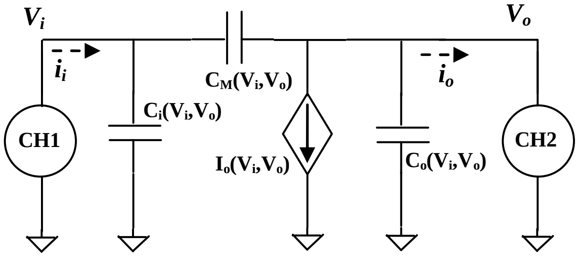


Figure 2. Characterization setup for a combinational logic cell.

Parasitic capacitances are pre-characterized through a series of transient simulations, in which saturated ramp input and output voltages are applied to input and/or output nodes while the output current is monitored. More specifically, to characterize $C_M(V_i,V_o)$, a saturated ramp is applied to CH1 and a DC voltage source is applied to CH2. Equation (1) is then simplified to:

$$i_o + I_o(V_i,V_o) - C_M(V_i,V_o)\frac{dV_i}{dt} = 0 \qquad (3)$$

In (3) C_M is the only unknown parameter for each V_{CH1} level of the input ramp and V_{CH2} of the output DC voltage source. As V_{CH1} changes (for example, from $-\Delta$ to $V_{DD}+\Delta$) and for a constant V_{CH2}, $C_M(V_{CH1},V_{CH2})$ values are calculated. The above experiment is repeated for each V_{CH2} between $-\Delta$ and $V_{DD}+\Delta$.

To characterize the output capacitance, $C_o(V_i,V_o)$, a DC source is connected to V_{CH1}, while a saturated ramp drives V_o. Equation (1) becomes:

$$i_o + I_o(V_i,V_o) + \left(C_o(V_i,V_o) + C_M(V_i,V_o)\right)\frac{dV_o}{dt} = 0 \qquad (4)$$

With already characterized C_M values, $C_o(V_i,V_o)$ is the only unknown parameter, which is easily calculated as before.

In our model C_o and C_M are dependent only on the input and output voltages. Therefore, according to this model, the slew of the ramp signal waveforms used in the transient analysis should not affect $C_M(V_i,V_o)$ and $C_o(V_i,V_o)$ values. However, in (3) for example, the dV_i/dt term represents the slope of the ramp signal applied to V_{CH1}, which may assume different values. If we change the input slew, the measured i_o value (for the same level of V_i and V_o) will also change. Fortunately, these two variations in dV_i/dt and i_o tend to counter each other so that change in the calculated $C_M(V_i,V_o)$ for different input slews is small. More importantly, the sensitivity of the output voltage waveform to $C_M(V_i,V_o)$ variation as a function of input slews is quite weak.

To be more precise, we have noticed that the $C_M(V_i,V_o)$ value can change for up to 5% for different input slews ranging from 50ps to 500ps whereas the change in output voltage waveform for the same range of input slews is only 0.2%. We have thus opted to ignore the effect of input slews on parasitic capacitance characterizations. In practice, we examine ramp signals with different slopes and use the average parameter values for all the ramps to fill up the lookup tables.

The following KCL equation is used to characterize $C_i(V_i,V_o)$, the parasitic capacitance seen at the input of a logic cell:

$$i_i - \left(C_i(V_i,V_o) + C_M(V_i,V_o)\right)\frac{dV_i}{dt} + C_M(V_i,V_o)\frac{dV_o}{dt} = 0 \qquad (5)$$

To characterize C_i, a DC source is connected to V_{CH2}, while a saturated ramp drives V_{CH1}, resulting in:

$$i_i - \left(C_i(V_i,V_o) + C_M(V_i,V_o)\right)\frac{dV_i}{dt} = 0 \qquad (6)$$

The only unknown parameter in this equation is $C_i(V_i,V_o)$, which is easily determined. This characterization is done for V_{CH2} values ranging from $-\Delta$ and $(V_{DD}+\Delta)$.

It is worth mentioning that the logic cell - characterization steps are performed independently of the load. As is well known, this is a major advantage of the current source modeling approach. The characterization steps for different combinational logic cells in a cell library are typically automated as part of a library characterization tool.

B. Output Voltage *Calculation*

A logical cell generally drives a circuitry including one or more logic cells through a short or long piece of interconnect. This whole circuitry can be considered as a load. Typical cell delay models are forced to model this load an effective capacitance to make the load compatible with the characterized cell library. However using our CSM, we have the advantage of using any type of load model to increase the accuracy of cell delay analysis. This loading effect should be considered for output voltage calculation. KCL at the output node results in the following equation:

$$i_L + I_o(V_i,V_o) + \left(C_o(V_i,V_o) + C_M(V_i,V_o)\right)\frac{dV_o}{dt} - C_M(V_i,V_o)\frac{dV_i}{dt} = 0 \qquad (7)$$

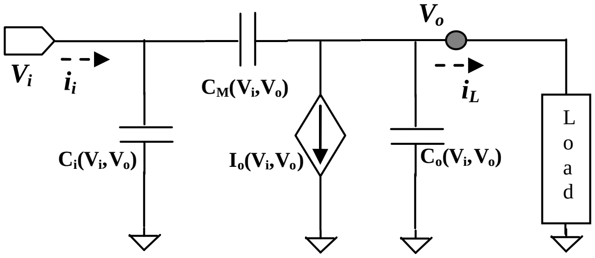


Figure 3. logic cell CSM and its load to calculate V_o .

iL in Equation (7) denotes the current drawn by the load. Note that Equation (7) is essentially the same as Equation (1) in which io has been substituted with iL. Note that i_L is the admittance function of the load multiplied by the output voltage. In [9] it is shown how to use the Pade method to approximate the admittance function of an RC network with a reduced order representation. As reported in [9] in most cases only one Pade term (i.e., Π model approximation) is sufficient for the error to be within

2-3% of Spice and in a few cases where one pole is not sufficient, more Pade terms should be preserved.

Given such an approximation (which need not be a single effective capacitance), one can numerically calculate the logic cell output voltage by any number of integration methods (In our implementation, we use the Euler Integration method [14]). Notice that iL is a dependent variable in terms of the output voltage, Vo, and the load.

The output voltage depends on the load, which is typically an RC network with capacitances for its sinks.

The CSM described above is used to model logic cells with a single Channel-Connected-Component (CCC) [15]. Examples of a single CCC are Inverter, NAND, and NOR cells. For the case of multi-stage logic cells, such as OR and AND gates, the logic cell should be divided into multiple CCCs. For each CCC, a CSM should be generated. For example, AND (i.e., a NAND followed by an Inverter) has two CCCs, therefore a cascade of two CSMs is used to model the AND gate. To calculate the output voltage of the AND cell, first the output voltage value of the NAND cell is calculated. This voltage is then input to the inverter cell to produce the output voltage of the AND cell.

III. CS MODELING – LATCHES

This section explains the CS model for a CMOS latch. As mentioned earlier, the CS model can be used to calculate the output voltage waveform given an input voltage waveform of arbitrary shape, including one with noise-induced glitches. These glitches can cause functional errors if they are latched into sequential cells.

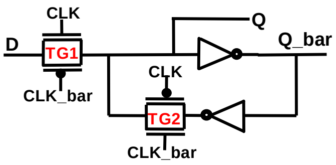
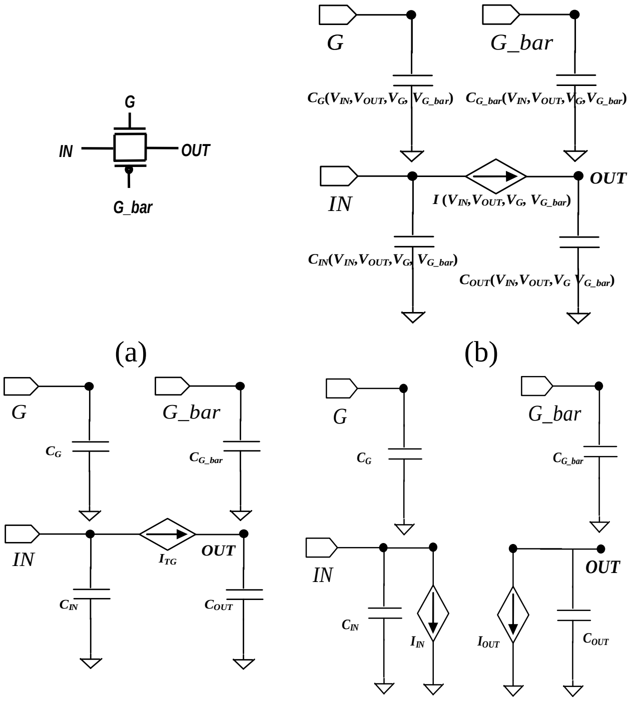


Figure 4. A positive level-sensitive CMOS latch.

Most sequential cells such as flip-flops and latches have at least one feedback loop to store a desired logic state [16]. As an example, Figure 4 shows a simple latch with a data input, D, a clock input, CLK, and true output, Q, and the complementary output Q_bar. The goal is to devise a CSM capable of computing the output voltage waveforms (for nodes Q and/or Q_bar) given the input voltage waveforms for data and clock nodes. The feedback loop is the most challenging part of such a model, because the noise which has been transferred to the output node through the path from the inputs to the output can be magnified and fed back to input. The model must be capable of accounting for this feedback-magnification effect.

We show how to construct CS models for specific instances of sequential cells (i.e., a transmission gate based latch and a master-slave flip-flop). This construction makes use of the circuit schematic of the flip-flop and requires understanding of the detailed operation of the flip-flop. The CSM construction process for other flip-flops (including, for instance, the monostable- or time-window-based ones), which is desirable from a practical viewpoint, has not been automated. Although this is an important undertaking, it falls outside the scope of the present paper.

We construct the CS model for a transmission gate (TG) which is commonly found in sequential circuit elements (cf. Figure 5(a)). The TG essentially acts as a non-linear resistor with the resistance value adjusted by its control input voltage (V_G and V_{G_bar}) as well as its input and output voltage levels. The nonlinear resistance behavior of the TG can effectively be modeled by a current source (cf. Figure 5(b)).


G
IN
OUT
G_bar
G
G_bar
C_G(V_IN,V_OUT,V_G, V_G_bar)
C_G_bar(V_IN,V_OUT,V_G,V_G_bar)
IN
I (V_IN,V_OUT,V_G, V_G_bar)
OUT
C_IN(V_IN,V_OUT,V_G, V_G_bar)
C_OUT(V_IN,V_OUT,V_G V_G_bar)
G
G_bar
C_G
C_G_bar
IN
I_TG
OUT
C_IN
C_OUT
G
G_bar
C_G
C_G_bar
IN
OUT
C_IN
I_IN
I_OUT
C_OUT

(a)

(b)

(c) (d)

Figure 5. (a) A transmission gate, (b) its 4-D CSM, (c) its 3-D CSM for node G (a similar model is used for G_bar, (d) the decoupled version of the 3-D CSM).

Each capacitance in Figure 5(b) models the parasitic effects seen at the respective node. There also exists the Miller effect between every two nodes. The corresponding Miller capacitance between every pair of nodes is decoupled and merged into the capacitance of each node.

It is necessary to consider the effect of both G and G_bar, therefore, the dependency becomes four dimensional (V_{in}, V_{out}, V_G, V_{G_bar}). However examining the TG closely, we see that the model components corresponding to the NMOS (PMOS) transistor do not depend on G_bar (G) voltage value. This makes all model components 3-dimensional, with each component dependent on V_{in}, V_{out}, and exactly one of V_G or V_{G_bar}.

The TG characterization setup is shown in Figure 6 and is performed in two steps: one with respect to node G, and the other with respect to G_bar. CH1 to CH4 are the voltage sources used during characterization. In the first step, G_bar (CH4) is forced to a HIGH voltage level to turn off the PMOS transistor while the NMOS transistor is characterized. Each component in this part is dependent on three voltage values, V_{in}, V_{out}, and V_G (CH1 to CH3, respectively.) The second step of the characterization is conducted similarly to model the PMOS transistor by forcing G (CH3) to a LOW voltage level, thereby, turning off the NMOS transistor. Each component in this part is dependent on V_{in}, V_{out}, and V_G_bar. To construct the complete model, the components of the afore-mentioned parts are combined as depicted in Figure 5(c).

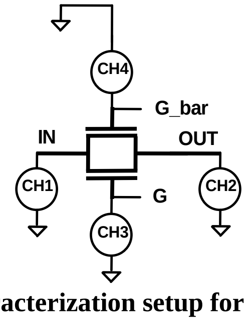


Figure 6. Characterization setup for a transmission gate.

The following set of equations defines the components in Figure 5(c):

$$\begin{aligned}
C_G &= C_G(V_{IN}, V_{OUT}, V_G)\\
C_{G\_bar} &= C_{G\_bar}(V_{IN}, V_{OUT}, V_{G\_bar})\\
C_{IN} &= C_{IN-G}(V_{IN}, V_{OUT}, V_G) + C_{IN-G\_bar}(V_{IN}, V_{OUT}, V_{G\_bar})\\
C_{OUT} &= C_{OUT-G}(V_{IN}, V_{OUT}, V_G) + C_{OUT-G\_bar}(V_{IN}, V_{OUT}, V_{G\_bar})\\
I_{TG} &= I_G(V_{IN}, V_{OUT}, V_G) + I_{G\_bar}(V_{IN}, V_{OUT}, V_{G\_bar})
\end{aligned} \tag{8}$$

where sets {C_G, $C_{IN\text{-}G}$, $C_{OUT\text{-}G}$, I_G} and {C_{G_bar}, $C_{IN\text{-}G\_bar}$, $C_{OUT\text{-}G\_bar}$, I_{G_bar}} represent the NMOS and PMOS model components, respectively. $C_{IN\text{-}G}$ and $C_{IN\text{-}G\_bar}$ are connected in parallel, and hence, they can be added into C_{IN}. Similarly, C_{OUT} and I_{TG} consist of their respective parallel-connected components as shown in Equation set (8).

The current source I_G can be decoupled into two current sources at the input and output ($I_{IN\text{-}G}$ and $I_{OUT\text{-}G}$, respectively.) Similarly I_{G_bar} can be decoupled into $I_{IN\text{-}G\_bar}$ and $I_{OUT\text{-}G\_bar}$. $I_{IN\text{-}G}$ and $I_{IN\text{-}G\_bar}$ are parallel with each other and can be added to I_{IN}. Similarly, $I_{OUT\text{-}G}$ and $I_{OUT\text{-}G\_bar}$ can simply be added into I_{OUT}. The resulting model with decoupled current sources is shown in Figure 5(d). Notice that similarly to what was done for current source characterization of combinational cells (cf. section II), the TG current sources are characterized using DC voltage sources. In addition parasitic capacitances are characterized through transient simulations. For example, for the C_{OUT} (C_{IN}) model components, a transition is applied to the output (input) voltage while the input (output) voltage is connected to a DC source.

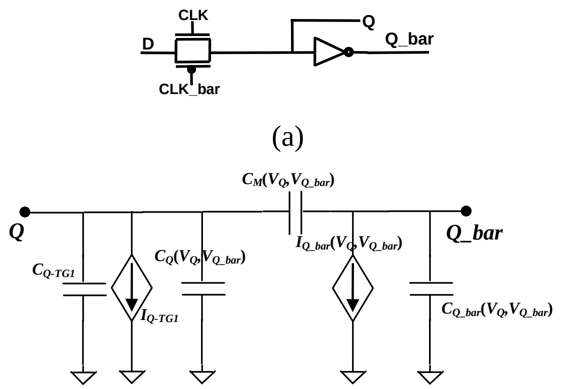
C. Mode-based Analysis of a Latch

At any time instance, the latch can be in one of the three modes: *transparent*, *opaque* (hold), or *transition*. In order to have an accurate CSM, the behavior of the latch in each mode should be investigated. In the following, we introduce the CSM for each mode. This step by step description helps up provide the intuition behind our complete model. Note that we will present a complete CSM (Figure 9) which covers all different modes and is able to adapt itself and calculate the output voltage in any mode.

i. Transparent mode (CLK=1)

In this mode CLK = 1 (and CLK_bar = 0), the latch is transparent, i.e., Q = D, TG1 is conducting while TG2 is OFF (cf. Figure 5). The inverter between Q and Q_bar passes the inverted D into Q_bar (cf. Figure 7(a)). The latch CS model in this mode can be obtained by connecting the CSMs for the inverter and

TG1 in series, resulting in the model depicted in Figure 7(b). Notice that the CSM model for TG has decoupled elements at its input and output as was shown in Figure 5(d). However, in Figure 7(b) we only show the output side of TG with components $C_{Q\text{-}TG1}$ and $I_{Q\text{-}TG1}$ (i.e., C_{OUT} and I_{OUT} in Figure 5(d)) because this is the only side for which we must write the KCL equations in order to calculate the Q and Q_bar voltage values. Recall that $I_{Q\text{-}TG1}$ is a voltage dependent current source which is dependent on the D voltage value. This is how the model captures the effect of input node (D) voltage.



(a)

$$I_{Q-TG1} = I_{Q-CLK\_bar}(V_D,V_Q,V_{CLK\_bar}) + I_{Q-CLK}(V_D,V_Q,V_{CLK})$$
$$C_{Q-TG1} = C_{Q-CLK\_bar}(V_D,V_Q,V_{CLK\_bar}) + C_{Q-CLK}(V_D,V_Q,V_{CLK})$$

(b)

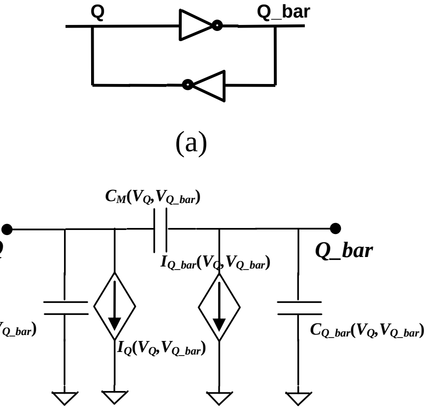
Figure 7. (a) Latch of Figure 4 in transparent mode, (b) Its CSM.

ii. Opaque mode (CLK=0)

In this mode CLK =0 (and CLK_bar=1), making TG2 conducting while TG1 is OFF. A feedback loop is thereby established such that the two inverters feed one another around the loop, while the input data is disconnected from the rest of the latch circuit (Figure 8(a)). The inverter model of Figure 1 is used back to back to construct the CSM for this case (Figure 8(b)). The scenario in which TG2 is partially conducting will be captured in the transition mode described below.

iii. Transition mode (CLK in transition)

This mode exists when CLK (CLK_bar) is making a falling (rising) transition and is not in the steady (high or low) state (e.g., when a setup or hold time test is performed.) In this case, the two TGs may be partially ON.



(a)

(b)

Figure 8. (a) Latch of Figure 4 in opaque mode, (b) Its CSM.

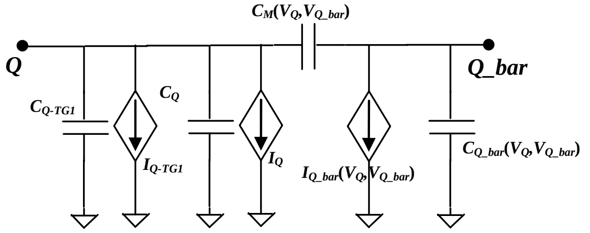
In contrast to the opaque mode where the feedback loop is closed and the two cross-coupled inverters are connected back to back, in the transition mode, the current to Q node through the feedback is controlled by CLK (CLK_bar). If CLK=1, this current will be zero; Otherwise, it will be equal the output current of the feedback inverter, i.e., I_Q in Figure 8(b). To account for this controlling behavior of the CLK/CLK_bar signals, we should make I_Q in Figure 8(b) dependent on those signals. We convert the 4-D CSM to a 3-D CSM, i.e., instead of using $I_Q(V_Q,V_{Q\_bar},V_{CLK}, V_{CLK\_bar})$, we utilize $I_{Q1}(V_Q,V_{Q\_bar},V_{CLK})$ and $I_{Q2}(V_Q,V_{Q\_bar},V_{CLK\_bar})$.

The transition mode must also work for the case when the feedback loop is open, i.e., CLK=1. In this case TG1 is conducting. Therefore, the CS model should be a superset of the CSMs in the transparent mode (Figure 7(b)) and the opaque mode (Figure 8(b)) with the I_Q made dependent on V_{CLK} and V_{CLK_bar}:

$$I_Q = I_{Q1}(V_Q,V_{Q\_bar},V_{CLK}) + I_{Q2}(V_Q,V_{Q\_bar},V_{CLK\_bar}) \qquad (9)$$

A similar situation applies to C_Q meaning that the parasitic capacitance at node Q is controlled by CLK and CLK_bar, i.e., two components of C_{Q1} and C_{Q2} are considered. The resulting model for the output nodes of the latch is presented in Figure 9. Note that node Q_bar is isolated from CLK and CLK_bar nodes by the inverter in the feedback loop; therefore, I_{Q_bar} values may be identified by dependency to Q and Q_bar only.

The CSM of Figure 9 can handle waveforms of arbitrary shapes at nodes D and CLK/CLK_bar inputs and enables construction of voltage waveforms at node Q and Q_bar for any operation mode of the latch.

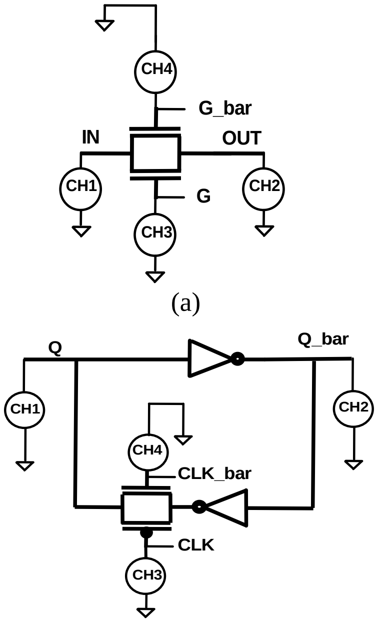


$$I_{Q-TG1} = I_{Q-CLK\_bar}(V_D,V_Q,V_{CLK\_bar}) + I_{D-CLK}(V_D,V_Q,V_{CLK})$$
$$C_{Q-TG1} = C_{Q-CLK\_bar}(V_D,V_Q,V_{CLK\_bar}) + C_{Q-CLK}(V_D,V_Q,V_{CLK})$$
$$I_Q = I_{Q1}(V_Q,V_{Q\_bar},V_{CLK}) + I_{Q2}(V_Q,V_{Q\_bar},V_{CLK\_bar})$$
$$C_Q = C_{Q1}(V_Q,V_{Q\_bar},V_{CLK}) + C_{Q2}(V_Q,V_{Q\_bar},V_{CLK\_bar})$$

Figure 9. CSM of the latch in transition mode.

D. *Pre-characterization*

We explain how to pre-characterize the CSM of Figure 9. The setup is shown in Figure 10. The latch is divided into two parts and each part is characterized separately. In the first step (Figure 10(a)) TG1 is characterized as explained earlier at the beginning of this section (c.f. Figure 6 and Figure 5.) and $C_{Q\text{-}TG1}$ and $I_{Q\text{-}TG1}$ are calculated. The second step of the characterization (see Figure 10(b) and the corresponding circuit model in Figure 11) is explained below.



(a)

(b)

Figure 10. Characterization setup for the CSM of Figure 9.

As stated earlier, I_Q is divided into I_{Q1} and I_{Q2} to reduce the dimension of characterization tables. To characterize I_{Q1}, CH4 is forced to zero while CH1 to CH3 voltage values are swept from -Δ to (V_{DD}+Δ). The current value sourced through CH1 is measured as $I_{Q1}(V_Q,V_{Q\_bar},V_{CLK})$. Characterization for I_{Q2} is done similarly. The characterization of I_{Q_bar} is only dependent on CH1 and CH2 values. By forcing these two supplies to a certain DC voltage level, the current sourced through CH2 will be unique regardless of the value of CH3 and CH4; therefore, there is no CLK and CLK_bar dependency for I_{Q_bar}.

To characterize C_M and C_{Q_bar}, we start from the KCL equation at the **Q_bar node** of the model in Figure 11:

$$i_{Q\_bar} + I_{Q\_bar}(V_Q,V_{Q\_bar}) - C_M(V_Q,V_{Q\_bar})\cdot\frac{dV_Q}{dt} + [C_M(V_Q,V_{Q\_bar}) + C_{Q\_bar}(V_Q,V_{Q\_bar})]\cdot\frac{dV_{Q\_bar}}{dt} = 0 \qquad (10)$$

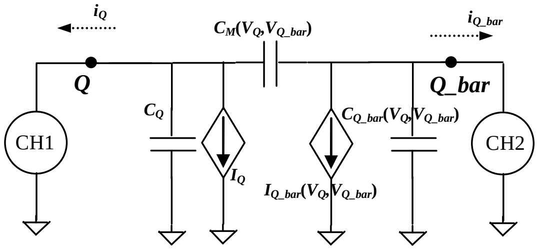


Figure 11. Circuit model of the characterization setup for Figure 10(b).

A number of transient simulations are performed to characterize the capacitive elements of our CSM. To pre-characterize the Miller capacitance C_M, a saturated ramp input voltage is applied to node Q (CH1 voltage source in Figure 11). Simultaneously, CH2 voltage value is swept from -Δ to (V_{DD}+Δ). The terms containing ${}^{dV_{Q\_bar}}\!/_{dt}$ in Equation (10) will thus be zero. Next, with the above setup, i_{Q_bar} (the current associated with CH2) is monitored. $I_{Q\_bar}(V_Q,V_{Q\_bar})$ is plugged into the equation for the corresponding voltage values of Q and Q_bar nodes. Since $I_{Q\_bar}(V_Q,V_{Q\_bar})$ has already been characterized, the only unknown parameter in Equation (10) is

$C_M(V_Q,V_{Q\_bar})$, which is thereby calculated. A similar procedure is used for the characterization of capacitance $C_{Q\_bar}(V_Q,V_{Q\_bar})$. However, this time a ramp voltage is applied to CH2 while CH1 is forced to DC voltage values.

As explained earlier for the combinational cell characterization, we have observed that the slope of the ramp input has a minor impact on the characterization results. As before, we examine ramp signals with different slopes and use the average parameter values for all the ramps to fill up the lookup tables.

To characterize C_Q, the KCL equation at the **Q node** of the model in Figure 11 is written as:

$$i_Q + I_Q + \{C_Q + C_M(V_Q, V_{Q\_bar})\} \cdot \frac{dV_Q}{dt} - C_M(V_D, V_{Q\_bar}) \cdot \frac{dV_{Q\_bar}}{dt} = 0 \qquad (11)$$

A saturated ramp voltage is applied to node Q (CH1) and CH2 is swept from -Δ to (V_{DD}+Δ). The term dV_{Q_bar}/dt becomes zero; C_M and I_Q are also known from the above-mentioned characterization steps. Therefore, C_Q can be calculated as a function of CH1, CH2 and also CH3 for its CLK-dependent component, i.e., C_{Q1}. Similarly its CLK_bar-dependent component (C_{Q2}) is calculated as a function of CH1, CH2, and CH4.

E. Output Voltage Calculation

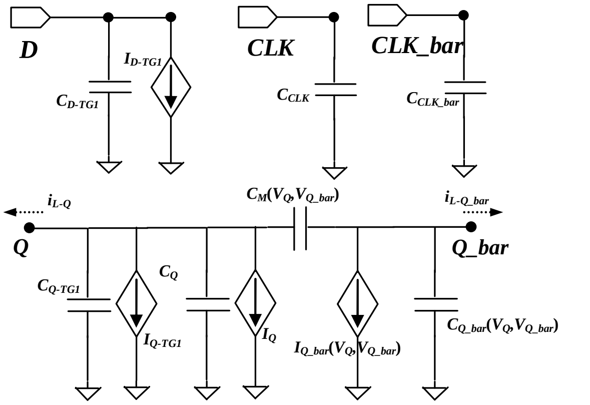
Figure 12 shows the complete CSM for the latch of Figure 4. The output voltage waveforms at nodes Q and Q_bar can be constructed for given input voltage waveforms (for CLK and D) in the presence of an arbitrary load. The following two KCL equations are used to calculate the voltage values at Q and Q_bar:

$$i_{L-Q\_bar} + I_{Q\_bar}(V_Q,V_{Q\_bar}) - C_M(V_Q,V_{Q\_bar}) \cdot \frac{dV_Q}{dt} + [C_M(V_Q,V_{Q\_bar}) + C_{Q\_bar}(V_Q,V_{Q\_bar})] \cdot \frac{dV_{Q\_bar}}{dt} = 0 \qquad (12)$$

$$i_{L=Q} + I_Q + I_{Q=TG1} - C_M(V_Q,V_{Q\_bar}) \cdot \frac{dV_{Q\_bar}}{dt} + \{C_Q + C_{Q=TG1} + C_M(V_Q,V_{Q\_bar})\} \cdot \frac{dV_Q}{dt} = 0 \qquad (13)$$

where $i_{L\text{-}Q\_bar}$ and $i_{L\text{-}Q}$ denote the currents drawn by loads at nodes Q_bar and Q, respectively. The effect of node D appears in $I_{Q\text{-}TG1}$ and $C_{Q\text{-}TG1}$. Voltage value of D is known because it is either a primary input or the output of some combinational cell, which means that its voltage has already been calculated based on the combinational CSM. As seen in Equations (12) and (13), the CSM components at nodes D, CLK, and CLK_bar are not required for the output voltage calculation; therefore, we do not explain the details of characterization for these components (although it can be done similarly to what explained in the previous section).

As before, a Pade approximation [9] is used to model the load and substitute $i_{L\text{-}Q}$ ($i_{L\text{-}Q\_bar}$) as a function of the output voltage V_Q (V_{Q_bar}.) An Euler integration method is used to numerically solve the two unknown voltages V_Q and V_{Q_bar} from Equations (12) and (13).



$$I_{Q-TG1} = I_{Q-CLK\_bar}(V_D,V_Q,V_{CLK\_bar}) + I_{D-CLK}(V_D,V_Q,V_{CLK})$$
$$C_{Q-TG1} = C_{Q-CLK\_bar}(V_D,V_Q,V_{CLK\_bar}) + C_{Q-CLK}(V_D,V_Q,V_{CLK})$$
$$I_Q = I_{Q1}(V_Q,V_{Q\_bar},V_{CLK}) + I_{Q2}(V_Q,V_{Q\_bar},V_{CLK\_bar})$$
$$C_Q = C_{Q1}(V_Q,V_{Q\_bar},V_{CLK}) + C_{Q2}(V_Q,V_{Q\_bar},V_{CLK\_bar})$$

Figure 12. (a) Complete CSM for latch of Figure 4.

Solving Equations (12) and (13) when there is no feedback for the sequential cell (i.e., CLK=1, CLK_bar=0, and so I_Q≈0), is similar to calculating the output voltage of an inverter for which the input comes from a transmission gate. When the feedback is present (i.e., CLK=0, CLK_bar=1, and so $I_{Q\text{-}TG1}$≈0), Equations (12) and (13)) update one another's current sources (I_Q and I_{Q_bar}), which thus models the magnification effect of the feedback loops. The other mode of operation, which is the transmission mode, is also captured by the dependency of $I_{Q\text{-}TG1}$ and I_Q on CLK and CLK_bar.

IV. CS MODELING – MS FLIP-FLOPS

An edge-triggered master-slave (MS) flip-flip comprises of two level sensitive latches: a negative level-sensitive and a positive level-sensitive latch. The first stage latch is referred to as the master latch while the second stage latch is called the slave latch. Figure 13 shows a positive edge-triggered flip-flop [16].

When CLK is low, the master negative level-sensitive latch output, Q_{M_bar}, follows the D input while the slave positive level-sensitive latch holds the previous value. When CLK makes a rising transition from 0 to 1, the master latch stops sampling the data input and holds the last data value at the time of the clock transitions (subject to setup time constraint). The slave latch becomes transparent, passing the stored master value of Q_{M_bar} to the output of the slave latch, Q. The D input is blocked from affecting the output because the master is disconnected from the D input. When CLK makes a falling transition from 1 to 0, the slave latch holds its last sampled value while the master starts sampling the input again.

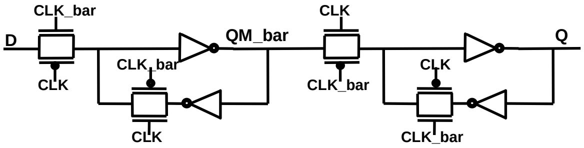


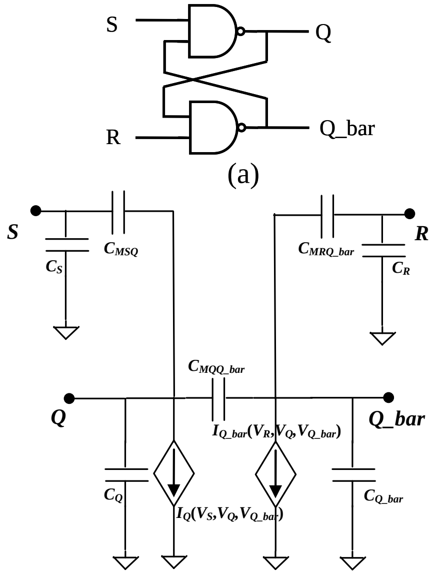
Figure 13. A positive edge triggered flip-flop.

To develop the CSM for a master-slave flip-flop, the latch CSM model of Figure 9 can be substituted for both the master and the slave latches. Therefore for a given input data and clock, the voltage values at Q_{M_bar} and Q can be calculated similar to approach in section III.E.

Since the two parts of the master-slave flip flop are not separated from each other and a transmission gate (which is a channel-connected component) is in between, the iterative approach should not separate the computation of V_{QM_bar} from V_Q and these computations should be performed simultaneously. In the experimental results section, we shall present the cases in which V_{QM_bar} and V_Q are iteratively and concurrently updated.

V. CS MODELING – SR LATCHES

In this section we briefly explain how the CS model for a different type of latch i.e., a SR latch can be created. Figure 14(a) shows an SR latch implemented using a pair of cross-coupled NAND cells. We use a multiple-input switching CSM for each NAND and then combine them to create the CSM for the SR latch. The resulting CSM is depicted in Figure 14(b).



(a)

(b)

Figure 14. (a) NAND-based SR latch, (b) CSM for SR latch.

The current sources at nodes Q and Q_bar are characterized by 3-D lookup tables. Although, in theory, capacitances at input and output nodes of the NAND are dependent on voltage values of the combinational cell terminals, these values are not as sensitive to these voltages as the non-linear current sources. Therefore, the number of entries in the capacitance look-up tables can be significantly smaller than that for the current-source look-up tables. The voltage values at Q and Q_bar can be calculated similar to section III.E.

VI. EXPERIMENTAL RESULTS

Our CSM simulator was implemented using C and Perl languages. All the experiments discussed in this section were performed on a Sun Fire V880 machine with the Ultra-SPARC III 750MHz processor running Sun Solaris operating system.

A. CSM evaluation for Combinational Cells

In order to show the effectiveness of our CSM for combinational logic cells, it was compared with Hspice. Waveforms of arbitrary shapes, ranging from a simple saturated ramp to crosstalk-induced noisy waveforms with voltage fluctuation as high as 85%-V_{dd}, were applied by using the setup of Figure 15. The set of experiments involved various logic cells, such as simple inverter and NAND gates, multi stage cells such as OR and AND, as well as complex cells such as AOI (And-Or-Invert).

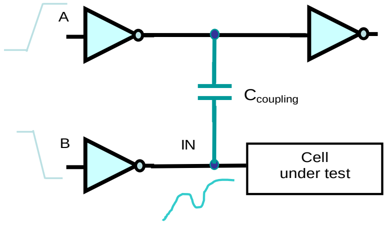


Figure 15. Experiment setup to create noisy waveforms.

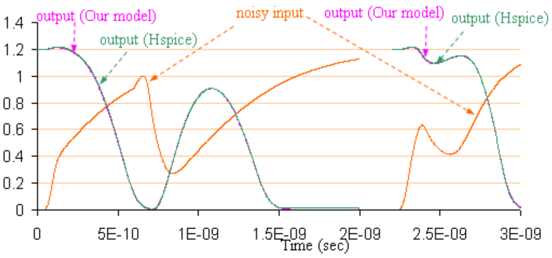
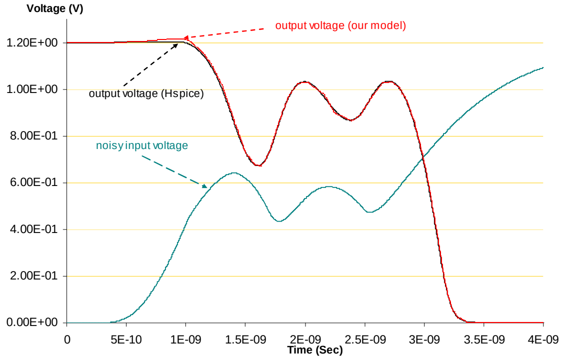


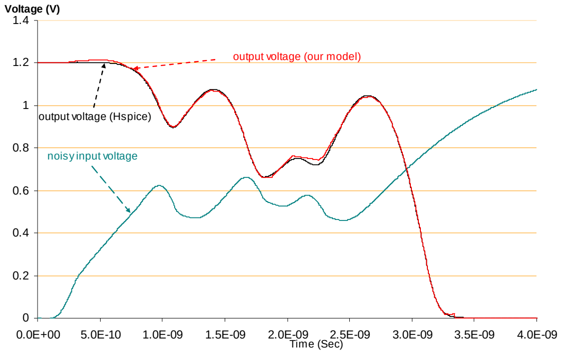
Figure 16. Hspice and CSM-produced waveforms for some crosstalk-induced noisy waveforms.

Figure 16 shows comparison with Hspice for some examples of crosstalk-induced noisy waveforms given to a minimum size inverter in our 130nm cell library. Figure 17 shows the comparison between our model and Hspice for different logic cells in the presence of multiple aggressors. The equivalent output waveforms generated by our model match those of Hspice very closely.

Next the accuracy improvement over one of the previous models, i.e., KTV [9] is discussed. For KTV, we made C_M and C_o constant and set them to the average value of their respective lookup tables. Figure 18 illustrates the absolute delay error comparison of our model and KTV with respect to Hspice for a minimum size inverter in our 130nm cell library. The input to the inverter is coupled by a 50fF coupling capacitance and is under attack by an aggressor net. Both the input of the inverter and the aggressor net are driven by minimum size inverters. The cell under consideration has a FO4 load. The signal arrival time at the input of driver line driver is set to 0ps while that of the aggressor driver (i.e., the noise injection time) is swept from 100ps to 200ps with a time step of 1ps. The slew values for the signal transition at the input of the victim and aggressor drivers are chosen from the range of 100ps to 500ps. This way we can create noisy waveforms of different shapes at the input of the inverter cell under interest. Compared to KTV, the accuracy of delay calculation for the minimum size inverter cell is improved by 8.8% (17.3%) in average (max.), respectively.



(a)



(b)

Figure 17. Comparison between our CSM and Hspice results for (a) minimum size inverter in a 130nm library given (double-aggressor) crosstalk induced noisy waveform (b) minimum size NAND3 in 130nm library given (triple-aggressor) crosstalk induced noisy waveform.

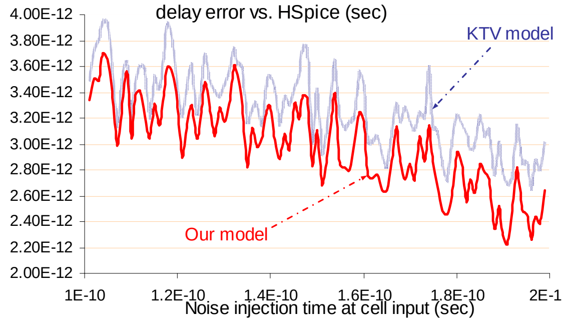


Figure 18. Absolute delay errors in delays calculated by our CSM compared to Hspice for an inverter of size x.

Figure 19 shows the absolute delay error trend for a similar experiment performed on AOI22 cell with size 10*x*, where *x* is the minimum size AOI22. The coupling value is 80fF and the arrival time of the aggressor line input driver is swept from 100ps to 250ps with time step of 1ps. The accuracy improvement in this case is 52.1% (93.4%) in average (maximum.)

The high accuracy of our model is mainly due to our accurate parasitic effect modeling during cell characterization, where the dependency of such effects to input and output voltage values are considered. In general, the error in 50%V_{dd} cell propagation delay is less than 0.7% (2.4%) in average (maximum) compared to Hspice for the cells in our 130nm library.

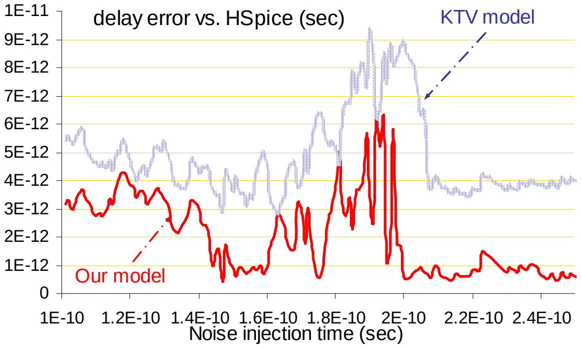


Figure 19. Absolute delay errors in calculated delays by our CSM compared to Hspice for an AOI22 gate of size 10x.

The shape of the waveform highly impacts the accuracy of timing analysis; therefore, delay and output slew metrics may not be sufficient to construct shape of the waveform. We use the Root Mean Squared Error (RMSE) as a metric to compare waveform similarities. RMSE is defined as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(V_{SPICE}\left(t_k\right) - V_{CSM}\left(t_k\right)\right)^2} \qquad (14)$$

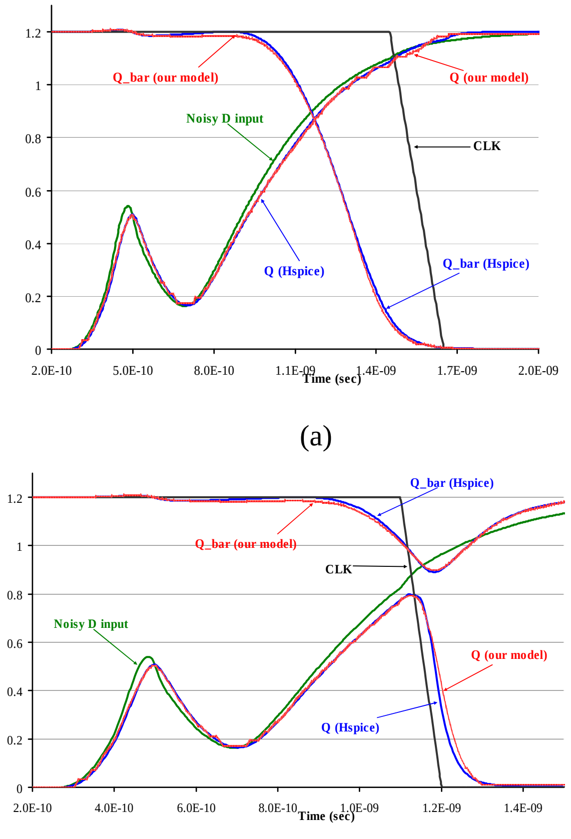
V_{SPICE} and V_{CSM} are the voltage values of the output of the logic cell at a given time. For each experiment, k=1 represents t_1 which is the time at which the noisy input starts to change whereas k=N represents t_N when V_{CSM} reaches its stable final value (either high or low). We finally normalize RMSE to V_{dd} to take out the effect of V_{dd} scaling. Note that RMSE has the same unit as the quantity being estimated which in this case is voltage. As mentioned earlier, the noise injection time shows the skew between the arrival of the aggressor and that of the victim signal transition. Table 1 reports the RMSE values for a few noise injection times for the combinational cells used in our experiments. It shows that our model is able to compute close-to-Hspice output waveforms in terms of their actual shape. Dimension of the CSM lookup tables are 33×33 for all the experimental results reported in this section. Our experiments showed that increasing the size of the tables to 66×66 increases the waveform similarity by up to 17%; we have chosen 33×33 size to achieve a reasonable tradeoff between result accuracy and runtime/memory efficiency.

TABLE 1. WAVEFORM SIMILARITY (NORMALIZED RMSE) COMPARISON WITH HSPICE FOR COMBINATIONAL LOGIC CELLS

Noise injection time (psec)		50	100	150	200	250
RMSE	Inverter	3.5e-3	4.4e-3	3.7e-3	3.1e-3	3.1e-3
	NAND2	4.3e-3	5.1e-3	7.1e-3	3.9e-3	4.4e-3
	AND2	5.1e-3	4.8e-3	5.9e-3	4.7e-e	4.6e-3
	OR3	6.2e-3	7.2e-3	5.8e-3	6.7e-3	6.2e-3
	AOI2	5.2e-3	4.3e-3	6.8e-3	5.1e-3	7.2e-3

B. CSM evaluation for Sequential Cells

To evaluate our CSM models for the sequential cells, we also use Hspice [13] to provide the “golden” result. In our experiments we considered voltage waveforms with arbitrary shapes from simple saturated ramps to crosstalk-induced noisy waveforms with voltage fluctuations as high as 85% of Vdd.



(a)

(b)

Figure 20. Crosstalk induced noise at the D input (a) noise changes state of the latch (b) it does not change state of the latch

It is important to capture the noise effects at the output of the latch to determine whether noise can flip the state of the latch through the feedback loop(s). Experiments were performed by using latches and flip-flops of different types (to be described later) comparing Q and Q_bar output waveforms with those of Hspice. An example of such experiments is depicted in Figure 20a). It is seen in this setup that although input D is noisy, the noise does not result in the change of state for the latch i.e., Q_bar changes from High to Low as expected. The Q and Q_bar waveforms generated by our model closely match the Hspice waveforms. Figure 20(b) shows another setup in which the noisy input of the latch has resulted in an illegal change of state. The Q_bar signal remains at High level and causes a functional error. Figure 20(b) shows how closely the latch output voltage calculated by our CSM model matches that in Hspice.

Similarly to what we did for the case of combinational cell models, we calculate the RMSE for the latch model to measure its waveform similarity to Hspice. Equation (14) is used to calculate the RMSE. However in this case, V_{CSM} represents the voltage values of the latch output (Q) at a given time. For each experiment, k=1 represents t_1 which is the time at which the noisy input D starts to change whereas k=N represents t_N when both Q and Q_bar reach their stable final values (either high or low). We finally normalize the RMSE to V_{dd} to take out the effect of V_{dd} scaling. To generate different noisy waveforms for D, the noise injection time (which is defined as the arrival time of the aggressor that is attacking the D signal) is swept from 100ps to 600ps with a step size of 5ps. Slew values for the signal transition at the input of the victim and aggressor drivers are in the 100ps to 500ps range. The CLK signal was kept fixed at 1.6ns. Note that in some of the cases unwanted change of the latch may occur.

Table 2 shows the normalized RMSE for some of these cases in 130nm library for the Q_bar output. Latch1 and Latch2 are transmission gate based latches (Figure 4) of different sizes (in latch1 all elements are minimum size and in latch2 they are all 10x) whereas latch 3 is a minimum size SR type latch (Figure 14). FF1 is minimum size a master-slave flip-flop as depicted in Figure 13. As reported in Table 2, the RMSE is around 1% of V_{dd}, which confirms that our voltage waveform closely matches that produced by Hspice.

TABLE 2. WAVEFORM SIMILARITY (NORMALIZED RMSE) COMPARISON WITH HSPICE FOR SEQUENTIAL CELLS

Noise injection time (psec)		200	300	400	500	600
RMSE	Latch 1	1.5e-2	.99e-2	1.1e-2	.81e-2	1.3e-2
	Latch 2	.87e-2	.74e-3	1.2e-2	.93e-2	1.1e-2
	Latch 3	1.2e-2	.77e-2	.94e-2	1.4e-2	1.6e-2
	FF1	.49e-2	.68e-2	.76e-2	.81e-2	.51e-2

To see the effect of technology scaling on the accuracy of our CSM, we performed some experiments by using Predictive Technology Model (PTM) [17] for 90nm and 65nm. For each cell and for each technology, we calculated the average normalized RMSE.

TABLE 3. WAVEFORM SIMILARITY (NORMALIZED RMSE) COMPARISON WITH HSPICE FOR DIFFERENT CELLS IN DIFFERENT TECHNOLOGY

Library	Cell	Average normalized RMSE		Runtime speedup vs. Hspice
		Q	Q_bar	
130nm	Latch 1	13.5e-3	11.1e-3	1220
	Latch 2	14.1e-3	12.2e-3	1220
	Latch 3	16.7e-3	13.5e-3	2130
	FF1	6.5e-3	7.3e-3	1110
90nm	Latch 1	12.5e-3	10.1e-3	1230
	Latch 2	14.5e-3	12.8e-3	1330
	Latch 3	17.1e-3	13.3e-3	2160
	FF1	6.9e-3	7.9e-3	1150
65nm	Latch 1	12.9e-3	10.6-3	1290
	Latch 2	14.7e-3	13.3e-3	1290
	Latch 3	17.3e-3	14.1e-3	2170
	FF1	7.6e-3	8.2e-3	1410

These results are reported in Table 3. In addition to sweeping the noise injection time from 100ps to 600ps, the CLK signal was also swept from 1ns to 1.9ns with a step size of 5ps. This resulted in 9000 different configurations for each cell under evaluation. It is seen that the CSM-based calculator is on average 1200 times faster than Hspice while producing results with nearly the same accuracy.

VII. CONCLUSIONS

An accurate current source model for combinational cell was presented. Furthermore, CSMs for sequential cells such as transparent latches and master-slave flip-flops were introduced. In addition to multi-stage logic nature of the sequential cells, the main challenge was the presence of feedback loops. Our proposed model addressed those, by creating the necessary current source and parasitic components. Given the input and clock voltage waveforms of arbitrary shapes, our model can accurately compute the output voltage waveform of a register cell, and hence, the timing and noise parameters associated with the cell. This was shown to considerably reduce the pessimism in timing and noise analysis. Experimental results for our current source sequential cell model demonstrate close-to-Hspice waveforms with significant runtime speedup.

REFERENCES

[1] D. Blaauw, V. Zolotov, S. Sundareswaran, "Slope propagation in static timing analysis," *Trans. Computer Aided-Design of Integrated Circuits & Systems*, pp. 1180-1195, 2002.

[2] D. Lee, V. Zolotov, D. Blaauw, "Static timing analysis using backward signal propagation," *Proc. Design Automation Conference*, pp. 664-669, 2004.

[3] V. Zolotov, and D. Blaauw, et al, "Noise Propagation and Failure Criteria for VLSI Designs," *Proc. of Int'l Conf. on Computer Aided Design*, pp. 587-594, 2002.

[4] N. Oh, L. Ding, A. Kasnavi, "Sequential cell noise immunity characterization using meta-stable point of feedback look," *Proc. Int'l Symp. On Quality Electronic Design*, 6 pages, 2006.

[5] Open Source ECSM Format Specification Version 1.2 September2005. http://www.cadence.com/webforms/ecsm.

[6] Composite Current Source (CCS) Modeling Technology Version 1.0 http://www.synopsys.com/cgi-bin/tapin.

[7] J.F. Croix, D.F. Wong, "Blade and razor: cell and interconnect delay analysis using current-based models," *Proc. Design Automation Conference*, pp. 386-389, 2003.

[8] V. Veetil, D. Sylvester, D. Blaauw, "Fast and Accurate Waveform Analysis with Current Source Models," *Proc. of Int'l Symp. On Quality Electronic Design*, pp. 53-56, 2008.

[9] I. Keller, K. Tseng, N. Verghese, "A robust cell-level crosstalk delay change analysis," *Proc. of Int'l Conf. on Computer Aided Design*, pp.147-154, Nov. 2004.

[10] P. Li and E. Acar, "Waveform independent gate models for accurate timing analysis", P*roc. Int'l Conf. on Computer Design*, pp. 363-365, 2005.

[11] C. Amin, C. Kashyap, N. Menezes, K. Killpack, E. Chiprout, "A nonlinear cell macromodel for digital applications," *Proc. of Int'l Conf. on Computer Aided Design*, pp. 678-685, 2007.

[12] H. Fatemi, S. Nazarian, M. Pedram, "Statistical Logic Cell Delay Analysis Using a Current-based Model," *Proc. of Design Automation Conference*, pp. 253-256, 2006.

[13] http://www.synopsys.com/products/mixedsignal/hspice/hspice.html

[14] Euler-Method, http://mathworld.wolfram.com/EulersMethod.html

[15] S. Sapatnekar, Timing, Chapter 4.2., Springer, 2004.

[16] N. H. E. Weste, D. Harris, "CMOS VLSI Design: A Circuit and System Perspective," Third edition, Addison-Wesley, 2005.

[17] Predictive Technology Model, http://www.eas.asu.edu/~ptm/