An Efficient Current-Based Logic Cell Model for Crosstalk Delay Analysis

Debasish Das, William Scott*, Shahin Nazarian*, Hai Zhou
EECS, Northwestern University, Evanston, IL 60208
*Magma Design Automation, San Jose, CA 95110

I. Abstract

Logic Cell modeling is an important component in the analysis and design of CMOS integrated circuits, mostly due to nonlinear behavior of CMOS cells with respect to the voltage signal at their input and output pins. A current-based model for CMOS logic cells is presented which can be used for effective crosstalk noise and delta delay analysis in CMOS VLSI circuits. Existing current source models are expensive and need a new set of Spice-based characterization which is not compatible with typical EDA tools. In this paper we present Imodel, a simple nonlinear logic cell model that can be derived from the typical cell libraries such as NLDM, with accuracy much higher than NLDM-based cell delay models. In fact, our experiments show an average error of 3% compared to Spice. This level of accuracy comes with an extra 15% runtime penalty in average compared to NLDM-based cell delay models.

II. 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 nonlinear behavior of logic cells is one of the main reasons which make crosstalk noise analysis challenging and CPU-intensive.

Due to inherent nonlinearity of driver cells, switch level static timing analysis techniques using simple resistive models are no longer applicable [1]. To address this issue, several researchers proposed current source models (CSMs) which model the gate nonlinear behavior with voltage controled DC current source and/or parasitic behavior with capacitances [2], [3], [4], [5]. But existing current source models are not compatible with the existing library models such as NLDM (Nonlinear Delay Model) and need Spice-based cell pre-characterization. On the other hand, at early stages of design using a CSM may be prohibitive and more efficient models are required. Imodel, our current-based logic cell model is developed to resolve the above shortcomings. Imodel, does not need any characterization and can extract the parameters from NLDM library. This advantage does not come with a big sacrifice and our experiments show crosstalk noise analysis with accuracy close to Spice.

III. Description of Imodel

First we need a model of current versus output voltage for a cell that has fully turned on. 3-parameter are used for this. For output rising, and voltage normalized to vdd, the model is

$$I_{on}(V) = I_{sat}(1 - \beta \times V - (1-\beta) \times V^{\alpha})$$

with $0 < B < 1$ and $2 < A < 6$. Figure 1 shows this curve compared to spice iv data for a buffer, rising and falling, and an 2-input nand gate, falling. This shows that this 3-parameter formula is capable of modeling the shape of real devices.

The relevant NLDM data is delay and output slew at smallest input slew and at large capacitance, so that the output transition is slow enough that the device can be assumed to be fully turned on. The key shape measurement extracted from this data is the ratio of the derivative of delay with respect to output load and derivative of slew

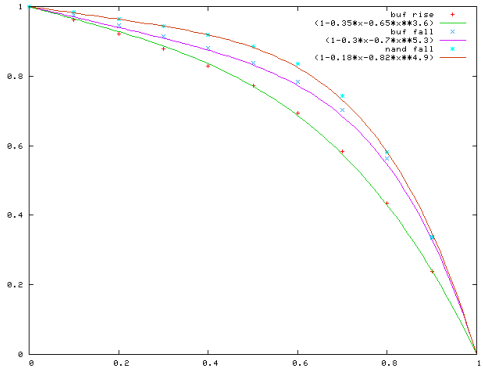

Fig. 1. $\frac{I_{on}(V_o)}{I_{sat}}$ compared to SPICE data

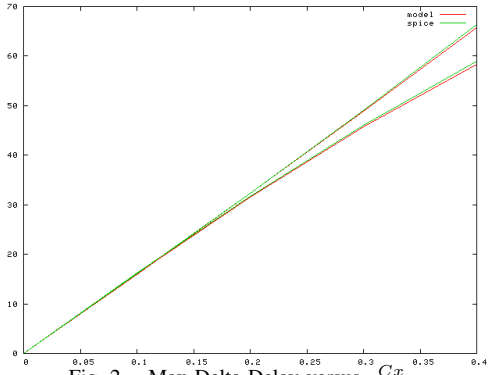

Fig. 2. Max Delta Delay versus $\frac{Cx}{Ctot}$

with respect to output load. It corresponds to the ratio of the integrals of 1/Ion(v) over the regions [0,0.5] and the [lo,hi] slew thresholds. But this is only one data point for the two paramters, α and β. In practice, if we have no other data, we set β=0.2 and use this ratio to determine A. We extract an output capacitance C_o by expecting that at large output load C and short input slew we should have output slew proportional to $C + C_o$. The largest two output load points in the output slew table are used for this. A parameter S is defined as switching time. The model becomes

$$I(v,t) = \begin{cases} I_{on}(V + 1 - t/S) & \text{if } 0 \le t \le S \\ I_{on}(V) & \text{if } t > S \end{cases} \quad (1)$$

When the model is driving very small capacitance, V will be approximately $\frac{t}{S}$. Accordingly, S is chosen so that the model matches output slew at low output load. Since the most effective aggressor alignment is often such that the aggressor is switching before the victim has begun to switch, and the victim's strength in this precharge region is determined by the opposite n/p type of transistor as determines the fully turned on current, we extract a conductance G_{hold} from the NLDM table with opposite rise/fall from the transition being modelled. The precharge current is

$$I(v,t) = -G_{hold} \times V \text{if } t < 0 \quad (2)$$

We now have a 6-parameter model where the parameteres are I_{sat},

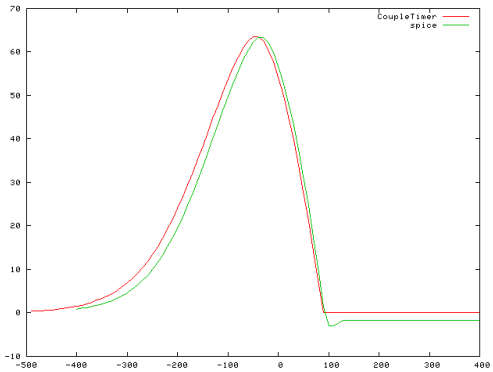

Fig. 3. Delta Delay versus Aggressor alignment

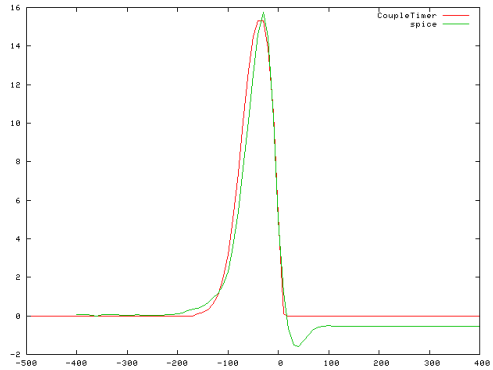

Fig. 4. Delta Delay versus Aggressor alignment

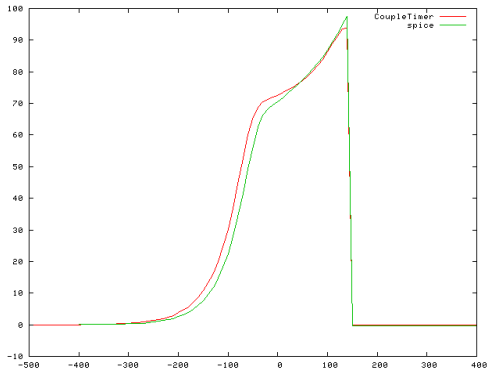

Fig. 5. Delta Delay versus Aggressor alignment

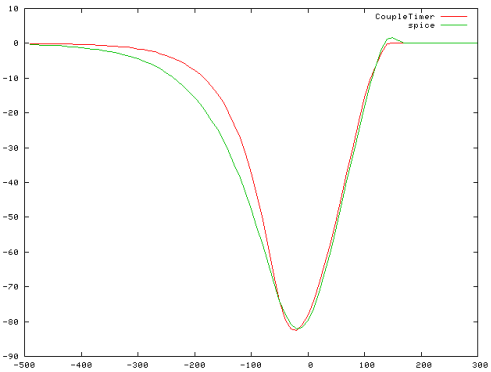

Fig. 6. Early Delta Delay versus Aggressor alignment

α, β, C_o, S and G_{hold}. However, this model is not too strong in the switching region, as if the competing n and p channels were fully on. So we multiply I(v,t) by a suppression factor $f(t) \leq 1$ in a region containing S. A default choice without introducing new parameters is a piecewise linear through the points $f(-S) = 1$, $f(0) = 0.2$, $f(S) = 0.2$, $f(2S) = 1$. This completes the description of the 6-parameter model that can be extracted from NLDM data. We did not add miller capacitance because we could not extract it effectively from NLDM data. Given the richer CCS and CCS-Noise libraries, miller capacitance should be added, and the turn on suppression factor $f(t)$ could be given additional parameters.

IV. Experimental Results

We focus on the case of a single aggressor. Both the victim and aggressor drivers are modeled as above. Aggressor alignment is optimized within the timing window. In the results presented here, there is no wire resistance, and we the circuit consists only of the two drivers, two grounded caps, and the coupling cap. Spice-like simulation is done with the $I(v,t)$ models, using the trapezoidal method and varying timesteps. The power V^{α} is approximated by a polynomial for speed. For Figure 2 C_x represents the coupling capacitance while C_{tot} is the sum of ground and coupling capacitance of the victim. Also X1 and X8 respectively represent smallest and largest size from cell library. Red lines in Figures 3-6 indicates our crosstalk aware static timer's output (CoupleTimer) while green line indicate SPICE simulation result.

• Crosstalk Delay vs $\frac{C_x}{C_{tot}}$

In Figure 2 we show the peak delta delay, for worst aggressor alignment in an infinite window, for an X1 buffer as victim, X8 buffer as aggressor, total victim cap = 40ff, total aggressor cap = 100ff, with varying coupling cap. The x-axis is the ratio of coupling capacitance to total victim capacitance.

• Crosstalk Delay vs Alignment curves

In Figures 3-6 we show crosstalk delta delay versus aggressor alignment, compared to SPICE. Figure 3 is late/slowdown delta delay for victim=BUFX1, aggressor=BUFX1, Figure 4 is for victim=BUFX1, aggressor=BUFX8. Figure 5 is for victime=aggressor= 2-input OR, X8. Figure 6 is early/speedup delta delay for victim=NANDX1, aggressor=NANDX8. One problem that we found with this model was poorer results for the case where the victim is a strong multi-stage device, and the aggressor is also strong. If a buffer, composed of two inverters, had minimal drive strength, then the two inverters may be roughly the same strength. But for a strong buffer, it is likely that the second inverter will be about 3 times stronger. This makes the internal node of the buffer sensitive to miller feedback from the output node. So as crosstalk changes the shape of the voltage waveform at the output, this causes a distortion in the waveform on the internal node. As long as the switching time and turnon form of the driver is derived from data taken from simulations of the device driving a simple load, it is hard to capture this effect.

V. Conclusions

Imodel, a current-based logic cell delay model was presented. Imodel has two main advantages: First is that unlike the existing current source models which need their own cell characterization data, Imodel does not need any Spice-based cell pre-characterization, and can extract the necessary information from the typical NLDM cell libraries. Second is the compatibility of Imodel with more accurate libraries, such as CCS library in order to create more accurate results. This makes Imodel flexible for different stages of circuit analysis and optimization where various accuracy vs runtime is desired.

References

[1] J. Qian, S. Pullela, and L.T. Pileggi. Modeling the effective capacitance of the RC interconnect. *IEEE Transactions on Computer-Aided Design of Integrated Circuits*, 13:1526–1535, December 1994.

[2] J.F. Croix and D.F. Wong. Blade and razor: cell and interconnect delay analysis using current-based models. In *DAC*, pages 386–389, 2003.

[3] I. Keller, K. Tseng, and N.K. Verghese. A robust cell-level crosstalk delay change analysis. In *ICCAD*, pages 147–154, 2004.

[4] H. Fatemi, S. Nazarian, and M. Pedram. Statistical Logic Cell Delay Analysis Using a Current-based Model. In *DAC*, pages 253–256, 2006.

[5] C.V. Kashyap, C.S. Amin, N. Menezes, and E. Chiprout. A nonlinear cell macromodel for digital applications. In *ICCAD*, pages 678–685, 2007.

[6] F. Dartu and L. T. Pileggi. Calculating worst-case gate delays due to dominant capacitance coupling. In *DAC*, pages 46–51, Anaheim, CA, June 1997.