

Timed Pattern Generation for Noise-on-Delay Calculation

Seung Hoon Choi
Dept. of ECE, Purdue University
West Lafayette, IN 47907
choi5@ecn.purdue.edu

Florentin Dartu
Strategic CAD Labs, Intel Corporation
Hillsboro, OR 97124
florentin.dartu@intel.com

Kaushik Roy
Dept. of ECE, Purdue University
West Lafayette, IN 47907
kaushik@ecn.purdue.edu

ABSTRACT

Computing the noise on delay effects is required for all circuits from simple ASIC designs to microprocessors. Transistor-level simulation engines make accurate delay calculation affordable only if the number of simulation per stage is very small. We propose a solution that predicts the alignment of aggressor signals with respect to the victim signal to induce the worst-case noise effect on delay. The aggressor alignment can be used to setup a detailed simulation. The worst-case delay in the presence of noise is predicted within 5% error for more than 99% of the cases tested using an average of 1.27 simulations per stage transition.

Categories and Subject Descriptors

B.8.2 [Hardware] : Performance Analysis and Design Aids

General Terms : Performance, Verification

1. INTRODUCTION

The practical solution to reducing the interconnect delay in deep sub-micron technologies by increasing the aspect ratio has an unfortunate side effect: the increased coupling capacitance to adjacent wires [1]. Therefore, the electrical signals in a circuit are increasingly affected by their layout neighbors. Crosstalk noise, the noise due to capacitive coupling, can affect circuit performance functionally and/or temporally. Functional failure is possible if the crosstalk-induced noise glitch is propagated and wrongfully evaluated. Delay effects are observed when two neighboring signals undergo switching at the same time. We will call the increase and the decrease of the delay due to crosstalk as *delay push-out* and *pull-in*, respectively. While the analysis of functional failure [2],[3] is very important for high speed circuits, more often the designers will have to deal with the other type of noise-activated problem: noise-on-timing effects. Our experiments with a large synthetic benchmark representative of a high-end microprocessor design environment showed that the coupling could increase the delay by 30% on average, making performance predictions a fortune-telling experience. This problem does not scale well for future process technologies as the timing margins are getting smaller [4]. The noise-on-timing scenario is described in Figure 1 for a gate with n aggressors.

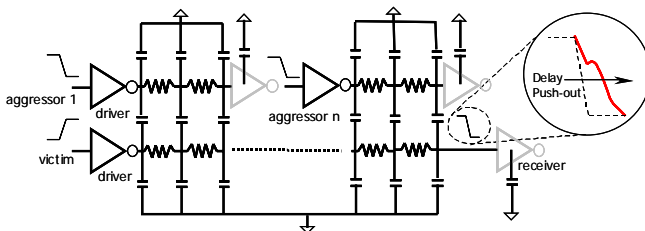


Figure 1 Noise-on-delay setup

This research was funded in part by MARCO GSRC under contract # SA3273JB and Intel Corporation.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 2002, June 10-14, 2002, New Orleans, Louisiana, USA.

Copyright 2002 ACM 1-58113-461-4/02/0006...\$5.00.

In static timing analysis (STA) [5], the circuit performance is estimated in a pattern independent manner where the delay calculation engine is called for every stage, sometimes multiple times, and therefore it has to be very efficient. To account for crosstalk in STA, the coupling capacitance is multiplied by a factor and modeled as grounded capacitance [6],[7]. The timing analysis is further complicated due to the circular aspects of the problem: aggressor switching affects the victim switching and vice versa [8],[9].

The worst-case delay and the corresponding signal alignment analysis based on a linear driver model [10],[11] has been very attractive for its simplicity although it can bring about inaccuracy. A new output resistance for the driver was proposed in [12] to improve the accuracy. However, it requires on-the-fly nonlinear simulations during analysis for gate modeling with cell level analysis. And at least four non-linear simulations are necessary to obtain the resistance value, which increases the run time significantly. In [12] a method is also proposed to align aggressors using the look-up table with given victim slopes, noise pulse widths and heights. Even though the table can be pre-characterized, the noise height and width should be extracted from the noise signal obtained by the iterative simulations explained above. Although [12] makes an important step toward a more accurate analysis, their priorities are sub-optimal considering that tool run time is very crucial.

We find the idea of transistor-level simulation for checking circuit performance very appealing. Many nonlinear effects are difficult to capture inside a simplified model. We propose to make assumptions in determining the signal alignment, and devoting the simulation cycles for the final verification. Our solution predicts the alignments of aggressors to result in the worst-case delay. The solution is the function of signal parameters for the noiseless case. We will show that by properly modeling the driver resistance, no nonlinear simulation is necessary to predict the alignment. Only one nonlinear simulation is necessary to obtain the delay with the predicted alignment. Once the initial delay values are obtained from the first simulation, we can test it to determine whether another run is needed to refine the alignments. The solution is tested with more than 8000 aggressor-victim combinations. The solution is extended to the delay pull-in. We show the results for both inverter drivers and gates that are more complex.

In section 2, delay push-out will be presented in more detail. In section 3, we describe our basic solution and explain the accuracy of the model. Improvements to the basic solution and the modification for pull-in are explained in section 4. Results are shown in section 5 and concluding remarks are in section 6.

2. DELAY PUSH-OUT

The signal whose delay is of interest is called a victim and all the other signals coupled to the victim line are considered aggressors. The delay is measured from the 50% crossing point of the voltage swing from a driver input to the 50% crossing point of a receiver

input in Figure 1. This way, both interconnect and gate delay can be considered at the same time. One victim can have more than one aggressor and we assume that one aggressor transition is not affected by the switching of other aggressors.

Figure 2 shows a typical delay variation of an inverter with a single aggressor. The x-axis represents the normalized slack - difference between aggressor and victim signal alignments divided by victim output transition time. Y-axis is the percentage delay increase. The sharp drop in push-out observable at the right side of this plot is an artifact of the delay definition using the 50% threshold. Because it is impractical to get an accurate delay profile like this via comprehensive search in a large circuit, an efficient way of doing it is essential to minimize the overhead.

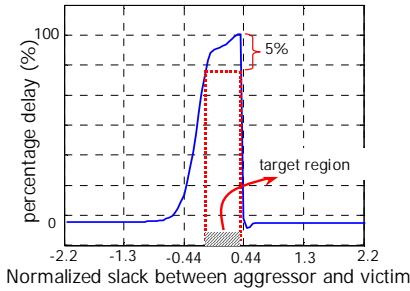


Figure 2 Delay variation vs. aggressor alignment

As it can be seen from the plot, the delay as a function of alignment is very difficult to model due to the large number of factors influencing the behavior. However, if we restrict the goal to finding out the alignment of aggressors to generate the delay within 5% of the worst case, the prediction problem becomes feasible. The solution that we provide will position the aggressor within the target region in Figure 2.

3. ALIGNMENT PREDICTION

3.1 Preliminary experiments

In [10], the victim driver was assumed linear and superposition was applied to obtain the composite noise waveform at the victim. This idea is depicted in Figure 3 for falling transition at driver output and rising transition for an aggressor. The "pushed-out" waveform is obtained by summing the "noiseless" waveform with the noise. The implicit aggressor alignment for worst-case delay requires the noise peak to occur when the "noiseless" transition crosses ($V_{CC}/2 - \text{peak}_v$). Proof to this is provided in [10],[11].

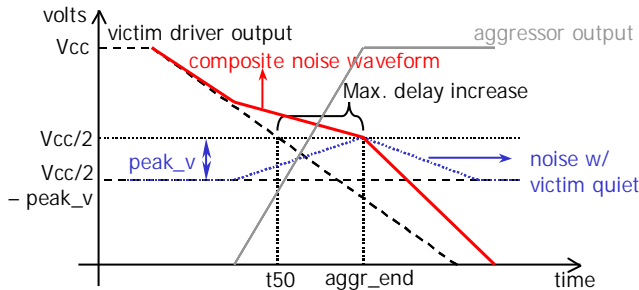


Figure 3 Worst-case delay and corresponding alignments

In order to attain some insights on how the alignment deviates from the linearity assumption in real circuits, inverter test-case circuits were generated and simulated. First, depending on the

interconnect lengths, three groups – short, medium and long – are selected. For each group, different combinations for aggressor-victim driver strengths – strong-weak, weak-strong and similar – are considered for two different metal layers. This way, 18 test cases are generated. These circuits are for only one type of signal switching direction. Then another circuit of different switching direction is added just to ensure that the aggressor alignment problem can be approached regardless of the switching direction. Now, the setup for 19 example circuits is completed to cover the possible combination of circuit parameters. There is only one aggressor affecting a victim for this set of experiments.

We decided to use the end-of-transition point as the time origin for the aggressor because for a single pole linear circuit assumption it represents the time when the noise will peak. Our goal is to find the timing relationship between the aggressor origin and the victim output transition for the worst-case alignment.

We first checked the relative timing position between the victim noiseless transition and the 5% max push-out window (target region) defined in Figure 2. The results are plotted in Figure 4 for the 19 examples. The x-axis is the normalized victim transition time such that the start-point and end-point correspond to zero and one, respectively and 0.5 is equivalent to the 50% crossing point, t_{50} . If the linearity assumption depicted in Figure 3 is correct, the max push-out should show up between 0.5 and 1 for noise with its peak smaller than $V_{CC}/2$ (as is the case with our circuits). However, the plot strongly implies that non-linearity plays an important part as most worst-case alignment fall outside this range. Some important aspects are to be extracted from the figure in the following subsection.

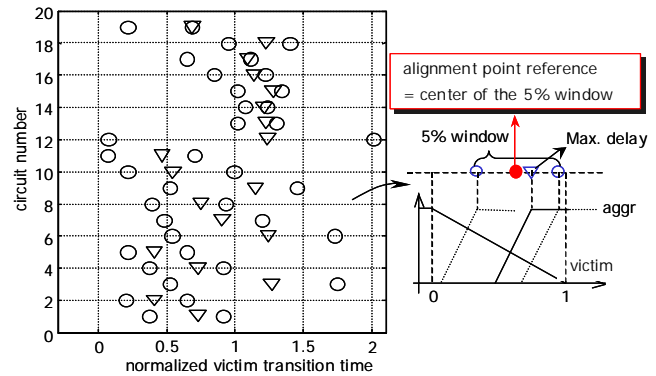


Figure 4 Aggressor alignments examples

3.2 Alignment prediction

Our primary goal is to correlate the intuitive understanding of the physics involved with these experimental results. A single number, $aggrAlign$, defined as follows, will represent each aggressor alignment:

Definition 1: $aggrAlign$ is the time when the aggressor transition ends, referenced to the beginning of the victim output transition and normalized to the victim's output transition time. (As an example, if the aggressor ends its transition in the middle of the victim output transition, then $aggrAlign$ is 0.5.)

The target being aligning the aggressors to have delay within 5% from the worst case, we will not even try to get the maximum push-out, in place we will try to match the middle point of the 5% window. This way we are obtaining the most robust solution.

The effect of crosstalk, which can be measured as the delay push-out, is dependent on each driver's strength, interconnect parameters, receiver size and the aggressor alignment. The driver strength combined with interconnect parameters and receiver size can be captured by looking at the driver output signal waveform, especially the transition time without coupling. The amount of coupling from an aggressor to a victim can be measured by the noise induced on a victim net when the victim is quiet. The same intuition can be applied when aligning the aggressor with slight variations. Basically, the aggressor alignment points within the normalized victim window, $aggrAlign$, is dependent on:

- $\frac{\text{aggressor driver output transition time}}{\text{victim driver output transition time}}$

As it can be seen from Figure 5, if the aggressor transition time is relatively larger than that of a victim, the $aggrAlign$ moves to the right implying larger $aggrAlign$. This is because $aggrAlign$ is defined as the aggressor switching end-point.

- peak noise, $peakV$, measured when the victim is quiet

Larger peak noise means the shift of aggressor alignment to the right, as shown in Figure 5. This holds true if we assume the driver is linear (see Figure 3).

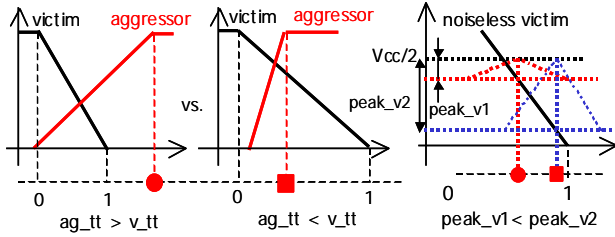


Figure 5 Shift of $aggrAlign$

To include these intuitive effects into a mathematical form we explored various empirical functions that preserve the theoretical relationships. The best model that we found is shown below.

$$aggrAlign = \alpha \cdot \frac{ag_tt}{v_tt} \cdot peakV^\beta + \gamma \quad (1)$$

Where $peakV$ is the peak of the noise waveform when victim is quiet and ag_tt and v_tt are the aggressor and victim driver output transition time without coupling, respectively. The fitting coefficients, α , β and γ , are obtained through a standard nonlinear regression [13], from the data of the 19 circuit examples for which the results were plotted in Figure 4. The procedure of applying the model to a circuit can be summarized as follows :

1. v_tt and ag_tt are measured at the victim and the aggressor receiver input, respectively. Both values can be obtained through nonlinear simulation or from regular characterization.
2. For multiple aggressors, align all the aggressors so that the noise peaks on a victim can also be aligned for max $peakV$. This assumes that the $peakV$ is obtained at the time when the aggressor reaches the end of its transition. This holds for low to medium resistive interconnect.
3. Run the non-linear simulation and measure the peak noise, $peakV$, at the receiver input node.

4. Insert ag_tt , v_tt and $peakV$ into (1) and obtain $aggrAlign$.
5. For each aggressor, align the input signal so that the output has its end-point at $aggrAlign$.

4. IMPROVING THE SOLUTION

In this section, several methods to improve the accuracy of the model and its extension to delay pull-in problem are presented.

4.1 Multiple Aggressors

If the results obtained using the procedure presented in Section 3.2 are grouped according to the number of aggressors, 0.80% of single-aggressor examples, 6.50% of two-aggressor examples and 9.56% of four-aggressor examples fail to align the aggressors for a delay within 5% of the worst case. This is because the coefficients were obtained through the regression with single-aggressor test cases. Thus, another set of coefficients for equation (1) can be obtained with multiple-aggressor test cases and used separately for multiple-aggressor cases to improve the accuracy.

4.2 Estimation of Noise Peak

For the procedure explained above, nonlinear simulations are needed for both $peakV$ and final delay computation. However, proper modeling of the nonlinear driver with linear resistance will allow us to skip the first simulation. Assuming ramp signaling for aggressor switching and replacing the transistor of a victim driver with a pre-characterized resistance, solving for the noise waveform on a victim driver output reduces to a linear system solution. The solution procedure is omitted here. By using this $peakV$ in the prediction procedure, we finally need only one nonlinear simulation to get the delay with the worst-case alignment.

4.3 Confidence of the Initial Prediction

Once the nonlinear simulation using the alignment defined by eqn. (1) is over, plenty of data is available and that can be utilized to further refine the results if necessary. The most suitable method that we found to test the quality of the results is to compare the noise peak time from the simulation with the 50% crossing point of the "pushed-out" waveform.

The actual peak noise is calculated by subtracting the noiseless waveform from the "pushed-out" waveform and taking the max value. In order to measure the noise at 50% crossing point, the 50%-point when the "pushed-out" noise waveform crosses $VCC/2$ is measured (T50 in Figure 6), and then the value is measured from the 'differential' noise at the T50. This procedure is described in Figure 6. For the delay to be the maximum, the peak noise and the noise at 50% should be the same assuming linearity as in Figure 3 [10].

The refinement procedure will look like this:

1. Determine if peak noise - noise at 50% > ϵ
2. If the previous condition is satisfied, then increase $aggrAlign$ by $\phi \cdot (\text{peak voltage} - \text{noise at 50\%})$

The parameters ϵ and ϕ can be optimally selected and this will trade off the accuracy and the number of additional simulations necessary. The reason that $aggrAlign$ is increased by the specified amount can be found in Figure 4. The alignment reference that is

the center of the 5% windows is usually located to the left of the worst-case alignment.

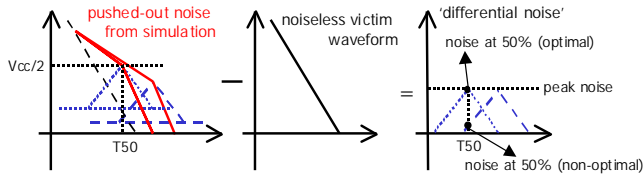


Figure 6 Decision on the reliability of the initial prediction

4.4 Delay Pull-In

When aggressor and victim signals are switching in the same direction, the delay of the victim will decrease. Accordingly, the alignment of an aggressor will be different from the one for push-out. However, the basic procedure used for the push-out problem can be applied with a little modification. From Figure 5 and eqn. (1), it is shown that $aggrAlign$ is dependent on $peakV$, so that β in eqn. (1) has positive value. But for the pull-in case, it becomes an opposite story. The larger $peakV$ is, the smaller $aggrAlign$ should be. This is verified when we get new coefficients fitted from pull-in test cases.

5. RESULTS

In order to check the accuracy of the model, we constructed a synthetic benchmark that is representative for Intel Corporation high-end microprocessor built on 0.18 μ m technology. The total number of circuits is 8159 and those are composed of inverters. The number of aggressors varies from one to four. The lengths of interconnect varies from 100 μ m to 2000 μ m. The transition time range used in experiments was the same as the range used in the microprocessor design. Tape out process parameters (transistor and interconnect) were used.

The accuracy of the model developed in Section 3.2 is shown in the second row of Table 1. The error was measured against the worst-case delay obtained by exhaustive search. Both delays at the input and output of the receiver are presented. The first two columns give the percentage of test cases for which the model was within the desired error range. The fourth and fifth columns show the mean and the standard deviation of the error for the entire benchmark, respectively. Using the basic model, out of 8159 cases, only 6.6% fail to locate the aggressors for delays within 5% of the worst case at the receiver input. Only 2% fail for the delay measured at the receiver output. The improvements introduced in Section 4 were also tested on the same benchmark and the results are also summarized in Table 1. The results for confidence check (section 4.3) reflect a second iteration applied selectively to 27% of the total circuits. The same percentage represents the run time overhead for this method. It can be deduced from the table that the usage of two sets of coefficients for multiple aggressor cases makes a difference. Replacing the victim driver with the resistance does not degrade the accuracy of the model. This is because $peakV$ is least sensitive to the variation in the model. If we are allowed to run another simulation with the confidence check, it is possible to align the aggressors to induce the delay within 5% from the worst case for more than 99% of the test cases. The last row of Table 1 represents the results for complex gates benchmarks which consists of 512 different circuits with NAND, NOR and AOI gates considering single-input switching.

The same fitting coefficients as the ones for inverter circuits are used to align the aggressor signals.

Table 1 Accuracy of the alignment solution

Delay at receiver input/output	within 3% (%)	within 5% (%)	Mean error (%)	Std. dev. error (%)
Basic model (Sec. 3.2)	84.2/91.1	93.4/98	1.59/1.06	1.98/1.30
Two models (Sec. 4.1)	92.7/96.5	98.7/99.7	1.23/0.71	1.13/0.94
Linear driver (Sec. 4.2)	92.7/96.6	98.5/99.6	1.21/0.67	1.15/0.95
Refining (Sec.4.3, $\epsilon=0.025, \phi=0.2$)	96.1/98.7	99.6/99.9	1.02/0.50	0.91/0.74
Complex gates	93.2/99.1	99.0/100	0.84/0.46	1.12/0.62

6. CONCLUSION

We approached the problem of aggressor alignment for the worst-case delay variation due to crosstalk. Our solution is useful when a transistor-level simulation engine is available during STA. Our goal was to use as few simulations as possible to determine the worst-case alignment. Due to the nonlinear simulations, the delay result has all the deep sub-micron electrical effects included.

Our model was obtained from the properties of the aggressor and victim signals and by scrutinizing their effects on timing. A simple analytical formula is used to map the signal parameters for the noiseless case into the aggressor alignment. The coefficients of this model are obtained once per design through a standard nonlinear fitting procedure. The model was also further improved for both runtime and accuracy. By using an estimated value of the peak voltage, we removed one simulation per gate while the accuracy was minimally reduced. We also introduced a convergence criterion for our algorithm. With a 27% runtime overhead, we were able to get 99.6% of the examples within 5% of the worst-case delay.

The alignment problem related to multiple-input-switching has to be considered for more generic cases. We consider this topic to be a natural extension of the work presented in this paper.

REFERENCES

- [1] M. Bohr, "Interconnect scaling—the real limiter to high performance ULSI," *International Electronic Device Meeting*, pp.241-244, 1995.
- [2] D. Somasekhar, et al, "Dynamic Noise Analysis for Precharge-Evaluate Logic Family," *DAC*, pp.243-246, 2000.
- [3] K.L. Shepard, et al, "Harmony: Static Noise Analysis of Deep Submicron Digital Integrated Circuits," *IEEE TCAD*, pp. 1132-1150, Aug 1999.
- [4] L. Chen, et al, "A New Gate Delay Model for Simultaneous Switching and Its Applications," *DAC*, pp.289-294, 2001.
- [5] R.B. Hitchcock, "Timing verification and timing analysis program," *DAC*, pp.594-604, 1982.
- [6] P. Chen, et al, "Miller Factor for Gate-Level Coupling Delay Calculation," *ICCAD*, pp.68-74, 2000.
- [7] A.B. Kahng, et al, "On Switch Factor Based Analysis of Coupled RC Interconnects," *DAC*, pp.79-84, 2000.
- [8] P. Chen, et al, "Switching Window Computation for Static Timing Analysis in Presence of Crosstalk Noise," *ICCAD*, pp.331-337,2000.
- [9] R. Arunachalam, et al, "TACO : Timing Analysis With Coupling", *DAC*, pp.266-269, 2000.
- [10] F. Dartu, L.T. Pileggi, "Calculating Worst-Case Gate Delays Due to Dominant Capacitance Coupling," *DAC*, pp.576-580, 1997.
- [11] P.D.Gross, et al, "Determination of Worst-Case Aggressor Alignment for Delay Calculation," *ICCAD*, pp.212-219,1998.
- [12] S. Sirichotiyakul, et al, "Driver Modeling and Alignment for Worst-Case Delay Noise," *DAC*, pp.720-725, 2001.
- [13] N.R. Draper, H. Smith, "Applied Regression Analysis," John Wiley & Sons, Inc., 1998.