Gate Delay Modeling for Static Timing Analysis of Body-Biased Circuits

Donkyu Baek, Insup Shin, and Youngsoo Shin
Department of Electrical Engineering
KAIST, Daejeon 305-701, Korea

Abstract—Body biasing is a well-known circuit technique to compensate for increasing process variations. A static body biasing, in which fixed amount of bias voltage is applied after chips are sorted out according to process corners, is a convenient and viable approach in ASIC designs. The key question in this approach is how we generate new gate delays without re-characterizing all the gates, which is expensive. We notice, for the first time, that the ratio of delay (of original gate and the gate after body biasing) from the time input starts to change to the time output crosses $V_{dd}/2$ can be regarded invariant, i.e. it is a function of input transition time and load capacitance, but not a function of gate type. Therefore, once the ratio is characterized using a sample gate such as an inverter, it can be used to derive delay of all the other gates. We also propose some refinement techniques to improve accuracy. Experiments with industrial 32-nm library indicate that the average error over re-characterization is about 4% with maximum error being 11%, when maximum body bias voltage is assumed. The errors decrease as smaller bias voltage is applied.

I. INTRODUCTION

Process variation is usually handled by resorting to process corners. The worst corner (WC) depicts the case in which devices are manufactured as the slowest, while devices are the fastest in the best corner (BC). As minimum feature size has scaled down, the gap between WC and BC has widened, e.g. the delay difference of an inverter at WC and BC is about 43% of its nominal delay at 45-nm technology, but it increases to 89% of nominal delay at 32-nm in our experiment. This is detrimental to ASIC design, because the design becomes more pessimistic to guarantee that it operates correctly at all process corners.

One of the solutions to this situation is body biasing [1]–[3]. Chips are sorted out after manufacturing to figure out the process corners that they belong to; forward body bias (FBB) is applied to the chips in WC and reverse body bias (RBB) is applied to those in BC. This effectively reduces the gap between WC and BC, because the chips in WC become faster due to FBB while those in BC become slower due to RBB, as Fig. 1 illustrates.

Designers can now assume that process corners are WC′ and BC′ instead of original WC and BC. This allows large saving in circuit area and power consumption [4]. The problem, however, is the lack of gate library under new process corners. Building a new gate library, which nowadays contains a thousand of gates, is too expensive; to make it worse, the amount of body bias may change for different designs.

In this paper, we propose a method that automatically generates a new gate library from an existing library at WC or BC. The key idea is that the ratio of delay (of original gate and body-biased gate) from the time input starts to change to the time output crosses $V_{dd}/2$ is characterized for a representative gate such as an inverter (for different input transition time and load capacitance), and can be used for all the other logic gates to derive their new propagation delay. Some refinements are necessary to improve accuracy for gates that have multiple stacks of transistors or consist of multiple stages, but the amount of extra effort is kept small.

The remainder of this paper is organized as follows. The key notations of gate delay are introduced in the next section. The main idea of gate delay modeling of body-biased circuits is addressed in Section III, which is followed by experimental validation in Section IV. The paper is summarized in Section V.

II. PRELIMINARIES

Body bias can be applied to both pMOS and nMOS transistors, even though the cost of application is different. Body bias of pMOS can be altered through applying a particular voltage to n-well; similarly, the voltage of p-well can be adjusted for body bias of nMOS, but this requires triple-well technology, which is more expensive. The body bias voltage is supplied through tap cells, which are placed in regular fashion in circuit layout [5]. In this paper, we consider the application of body bias to pMOS alone, but only for simplicity of presentation. Notice that, in this setting, rising transition is affected but falling transition is not.

Consider an inverter shown in Fig. 2. The propagation delay, denoted by $d_p$, is measured between the time when input

\[d_p = \frac{V_{dd}}{2} \times \frac{1}{c_{in}} \quad \text{for rising transition,}\]

\[d_p = \frac{V_{dd}}{2} \times \frac{1}{c_{out}} \quad \text{for falling transition,}\]

where $c_{in}$ and $c_{out}$ are input and output capacitance, respectively.

Fig. 1. Original process corners (WC and BC) and process corners after body biasing is assumed (WC′ and BC′).
voltage $V_i$ and output voltage $V_o$ become $V_{dd}/2$, respectively. We define $d_{0.5}$ as the delay from the time when $V_i$ starts to change to the time when $V_o = V_{dd}/2$. If the input transition before and after $V_i$ reaches $V_{dd}/2$ is symmetrical,

$$d_{0.5} = d_p + \frac{t_i}{2}$$  \hspace{1cm} (1)

holds, where $t_i$ is input transition time.

The propagation delay $d_p$ is a function of input transition time $t_i$ and load capacitance $C_L$. There are two different methods for its modeling: empirical expression containing $t_i$ and $C_L$ as parameters, e.g. k-factor equation and nonlinear delay model; 2-D lookup table indexed by $t_i$ and $C_L$. We assume the latter method in this paper, but without loss of generality.

III. GATE DELAY MODELING

A. Basic Concept

Consider the inverter shown in Fig. 2 again. We want to derive an analytic expression of $d_{0.5}$. The load capacitance $C_L$ is charged from 0 to $V_{dd}/2$ by the saturation current through pMOS, which is approximated by

$$I_d(t) \approx \alpha W (V_{gs}(t) - V_t),$$  \hspace{1cm} (2)

where $\alpha$ is a constant, $W$ is a channel width, and $V_t$ is a threshold voltage. The gate to source voltage $V_{gs}$ is given by

$$V_{gs}(t) = \begin{cases} -\frac{V_{dd}}{t_i} t, & 0 \leq t < t_i \\ -V_{dd} & t_i \leq t \end{cases}$$  \hspace{1cm} (3)

where falling ramp is assumed for $V_i$ and $t = 0$ is assumed to be the origin of ramp function.

We require that

$$\int_{t_1}^{d_{0.5}} I_d(t) dt = \frac{C_L V_{dd}}{2},$$  \hspace{1cm} (4)

where $t_1$ is the time when pMOS is turned on, therefore $t_1 = -V_i t_i/V_{dd}$ because of $V_{gs} = -V_i$. Substituting (2) and (3) into (4), and then solving (4) for $d_{0.5}$ yields

$$d_{0.5} = \begin{cases} -\frac{V_{dd}}{t_i} + \frac{t_i}{\alpha (W/C_L)} \frac{V_{gg}}{V_{dd}} & d_{0.5} < t_i \\ \frac{t_i (V_{dd} - V_i)}{2V_{dd}} + \frac{V_{dd}}{2 \alpha (W/C_L) (V_{dd} + V_i)} & t_i \leq d_{0.5} \end{cases}$$  \hspace{1cm} (5)

If body bias is applied to pMOS transistor, its $V_i$ will change to a new value $V_i';$ $d_{0.5}$ will also change accordingly and is denoted by $d_{0.5}'.$ For fixed values of $V_{dd}$, $V_i$ and $V_i'$ (for known body bias voltage), it is easy to see that the ratio between $d_{0.5}'$ and $d_{0.5}$, denoted by $\eta$, is only a function of $t_i$ and $W/C_L$, i.e.

$$\eta(t_i, W/C_L) = \frac{d_{0.5}'}{d_{0.5}}.$$  \hspace{1cm} (6)

From (1) and (6), we obtain the propagation delay when body bias is applied:

$$d_p' = d_{0.5}' - \frac{t_i}{2} = \eta \left( dp + \frac{t_i}{2} \right) - \frac{t_i}{2}.$$  \hspace{1cm} (7)

In a practical gate library, the number of values of $t_i$ and $W/C_L$ used for 2-D lookup table is quite small. Therefore, once a table of $\eta$ is characterized using a sample gate such as an inverter, it can be used to derive $d_p'$ of all the other gates by using (7). Consider delay lookup tables of three gates illustrated in Fig. 3. Each entry of table corresponds to $d_p'$ (i.e. original delay when body bias is not applied) for particular values of $t_i$ and $C_L$. The same value of $\eta$ can be used to derive $d_p'$ corresponding to three shaded entries, because their $t_i$ and $W/C_L$ are the same. Therefore, a single $\eta$ table of 3 by 3 entries can produce $d_p'$ tables of all three gates.

B. Refinement

1) Multiple Stacks: In Section III-A, we ignored the dependence of threshold voltage on gate type. Consider NOR2 gate shown in Fig. 4(b). When $M_2$ charges load capacitance, its effective threshold voltage $V_{tt}$ is different from that of pMOS in basic inverter in Fig. 4(a) due to the non-zero voltage drop across $M_1$, which is turned on; $V_{gs}$ is not exactly given by (3) for the same reason. When $M_1$ charges load capacitance this time, it also charges intrinsic capacitance between $M_1$ and $M_2$, thus does not exactly follow the same model presented in Section III-A.
Thus, in (12) we only characterize the gates using SPICE; they serve as reference of comparison.

For the proposed method, 4 tables of \( \eta \) were built through SPICE simulation with INV, NOR2, NOR3, and NOR4 being used as sample gates; and 116 arrays of \( k \) (each containing seven entries) were extracted. Note that \( k \) is obtained by very simple calculation (13).

### A. Assessment of Gate Delay Modeling

#### 1) Single-Stage Gates

Fig. 6 reports the maximum and average percentage errors of 61 single-stage gates. If rising ramp is applied to \( V_i \), the delay of INV is affected by body bias this time:

\[
\begin{align*}
\eta_p' & = \eta_p + \eta_p'' \\
& = (d_p - d_p'') + \eta_p'' \\
& \approx d_p - (1 - \eta)kC_L
\end{align*}
\]

where the approximation of \( d_p'' \) is used to obtain (16). To determine the value of \( \eta \) (recall that \( \eta \) consists of 2-D table), we need a signal transition time of \( V_{o1} \), which is unknown; we use a signal transition time of \( V_i \) (i.e. \( t_i \)) instead as an approximation, i.e. we approximate that signal transition time of \( V_i \) and \( V_{o1} \) is the same.

### IV. VALIDATION

Experiments were performed to validate the proposed gate delay modeling; a total of 103 combinational gates from industrial 0.9 V 32-nm library were used for test; the original propagation delay of each gate is given as a 7 by 7 table. FBB of 0.35 V was assumed for pMOS; two propagation delays (for body-biased gates) were compared to assess the accuracy:

- New delay tables are obtained by re-characterizing all the gates using SPICE; they serve as reference of comparison.
- New tables are automatically generated by using the proposed method.

The accuracy may be improved if we maintain different \( \eta \) tables for different number of pMOS stacks, i.e. one table for a single pMOS stack such as inverter and NAND2 gate, two tables for the gates in which two pMOS transistors are stacked such as NOR2 gate, and so on. Since the maximum number of stacked transistors is 3 or 4 in typical gate library, the number of \( \eta \) tables that need to be characterized is 6 or 10. Another simplification can be made by using the same \( \eta \) table for \( M_1 \) and \( M_2 \) in Fig. 4(b) together; total number of tables is then reduced to 3 or 4, which is our approach in experimental validation presented in Section IV.

### 2) Multiple Stages

Consider AND2 gate shown in Fig. 5. Its propagation delay consists of two components

\[
d_p = d_{p,1} + d_{p,2}
\]

corresponding to the propagation delay of NAND and INV, respectively; note that \( d_{p,1} \) and \( d_{p,2} \) are unknown and only \( d_p \) is characterized.

Suppose that body bias is applied, but only to pMOS transistors. If the input \( V_i \) receives a falling ramp, \( d_{p,1} \) is affected by body bias but \( d_{p,2} \) is not (if we ignore the change in \( d_{p,2} \) due to the change in signal transition time of \( V_{o1} \)). Thus,

\[
d'_p = d'_{p,1} + d'_{p,2}
\]

\[
= \eta (d_{p,1} + \frac{t_i}{2}) - \frac{t_i}{2} + d_{p,2}
\]

We now approximate \( d_{p,2} \) by using a linear model:

\[
d_{p,2} = k(t_i)C_L
\]

where \( k \) is a proportionality constant, and takes a different value for different \( t_i \). Substituting (8) for \( d_{p,1} \), and then (11) for \( d_{p,2} \) in (10) yields

\[
d'_p = \eta \left( d_p - kC_L + \frac{t_i}{2} \right) - \frac{t_i}{2} + kC_L.
\]

Notice that the same \( \eta \) introduced in Section III-A is used in (12). We only characterize \( k(t_i) \) for each gate type, which is done by simple interpolation. Since \( d_p = d_{p,1} + kC_L \), two propagation delays \( d_{p,A} \) and \( d_{p,B} \) are extracted from delay table at two different load capacitance \( C_{LA} \) and \( C_{LB} \), respectively. Subtracting two corresponding expressions and solving for \( k \) yields

\[
k = \frac{d_{p,A} - d_{p,B}}{C_{LA} - C_{LB}}.
\]

The process is repeated for different values of \( t_i \).
D makes a falling transition while three other inputs are tied to 0. AOI31 is electrically equivalent to NOR2 except that the width of pMOS transistor in upper stack effectively becomes three times that of NOR2; this causes relatively larger error.

Nevertheless, average error shown in Fig. 6(b) is small, ranges from 0.7% to 4.1%.

2) Multiple-Stage Gates: Fig. 8 reports the maximum and average percentage errors of 42 multiple-stage gates. The range of errors is quite similar to that of Fig. 6: maximum error is between 1.7% and 11.0% and average error is between 0.5% and 4.6%. A few approximations have been made in Section III-B.2, but there are two main sources of error:

- Signal transition time of \( V_{o1} \) is assumed to be the same as that of \( V_i \) (i.e., \( t_i \)) in (16).
- Propagation delay of the second stage is approximated by using a simple liner model (11).

The first source causes an error when both \( t_i \) and \( C_L \) are large (see Fig. 8(a)). Even if \( t_i \) is large, signal transition time of \( V_{o1} \) can be small (CMOS is regeneration logic in essence). Thus, if we use the value of \( \eta \) corresponding to \( t_i \), it causes an error in the computation of the delay of second stage \( d_{p,2} \); this can be severe when \( d_{p,2} \) is a major portion of overall delay \( d_p \), which is when \( C_L \) is large.

The error when \( t_i \) and \( C_L \) are small is caused by the second source. As \( C_L \) decreases, the error of (11) increases since intrinsic delay, which is invariant and is dropped from (11), becomes more important. As \( t_i \) decreases, this error becomes more profound because the delay of the first stage becomes smaller, which makes the delay of the second stage more important.

The accuracy of the proposed method has been assessed using individual gates, but the impact on overall circuit timing such as path delay and timing slack needs more analysis.

ACKNOWLEDGMENT

This work was supported by Samsung Electronics. The authors would like to thank Dr. Hyung-Ock Kim, Dr. Jun Seomun, Mr. Jaehan Jeon, Dr. Jung Yong Choi, Dr. Hyo-Sig Won, and Dr. Kee Sup Kim for helpful discussion.

REFERENCES