Journal of Circuits, Systems, and Computers
Vol. 22, No. 8 (2013) 1350072 (16 pages)

DOI: 10.1142/S0218126613500722

ACCURATE GATE DELAY EXTRACTION FOR TIMING ANALYSIS OF BODY-BIASED CIRCUITS*

DONKYU BAEK, INSUP SHIN and YOUNGSOO SHIN†

Department of Electrical Engineering, KAIST,
Daejeon 305-701, Republic of Korea
†*youngsoo@ee.kaist.ac.kr*

Received 15 March 2013
Accepted 17 July 2013
Published 4 September 2013

Static body biasing is a circuit technique in which bias voltage is selected from more than one available voltage after manufacturing. It allows circuits to be designed at more favorable process corners; but effective application requires gate delays to be available for the new process corners, without the expense of re-characterizing individual gates. We show that the new delay of a gate (when body bias is applied) can be extrapolated from its old delay without body bias together with old and new delays of a few reference gates. Output transition time, which is another component of gate timing model, is extrapolated in a similar manner. Experiments with an industrial 32-nm gate library show that the average error in the new gate delays is less than 4.3%.

Keywords: Body biasing; static timing analysis.

1. Introduction

Die-to-die process variation is handled by using process corners. The worst corner (WC) occurs when all the devices in a manufactured circuit run at their slowest; and the best corner (BC) occurs when the devices all turn out to run as fast as manufacturing tolerances permit. As technology has advanced, and devices have become smaller, the gap between a typical WC and BC has widened. For example, we measured the difference between the delay of an inverter at the two extreme process corners: it was 43% of the nominal delay in 45-nm technology, increasing to 89% in 32-nm technology.

Countering the gap between the extreme corners using circuit techniques is important to reduce pessimism of design, which is typically performed only at those corners, e.g., setup check at WC and hold check at WC and BC. Body biasing is one such technique. It involves sorting chips after manufacturing, measuring their

*This paper was recommended by Regional Editor Tongquan Wei.

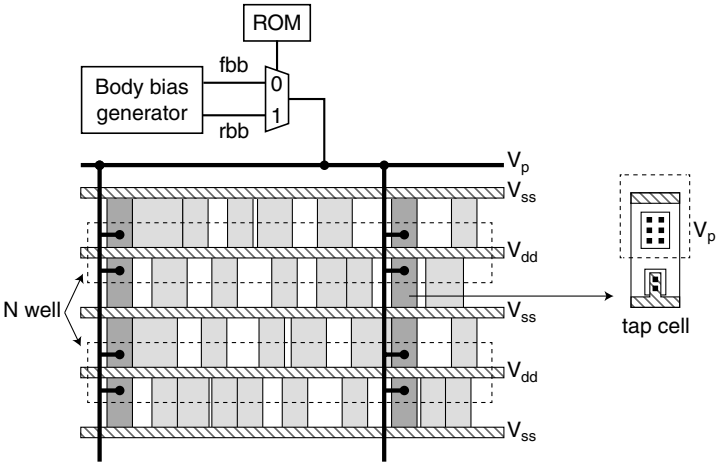


Fig. 1. Architecture of static body biasing. In this diagram only the pMOS transistors are biased.

leakage current or circuit delay, and assigning them to the appropriate process corners. As shown in Fig. 1, a chip classified on WC can then receive forward body bias (FBB) by writing 0 into a ROM; for a BC chip, reverse body bias (RBB) is selected by writing 1 into the ROM. The bias voltage is applied through regularly spaced tap cells.[1] This type of body biasing may be called static because the bias voltage that is applied is permanent[2–5]; whereas dynamic biasing techniques[6] change the voltage while the circuit is operating.

The use of body biasing relieves the designer from the necessity of considering the extreme process corners WC and BC. When the bias voltages, fbb and rbb in Fig. 1, are set, new process corners WC′ and BC′ are determined, and can be used during design. It is apparent that this form of body biasing can lead to large savings in circuit area and power consumption.[7] The drawback is the absence of a gate library for the new process corners. Building a new library from scratch is too expensive; furthermore, bias voltages may change for different designs and requirements.

There are several methods to build new library without expensive re-characterization. Fitting method[7] can estimate gate delay for any body bias voltage using delay values with zero bias. But, authors do not consider input transition time when they estimate gate delay. Output current modeling method[8] translates output current fluctuation due to body bias into modifications of load capacitance and input transition time. This method allows body biased gate delay to be estimated without additional library characterization. However, output current model during input transition is not accurate: average 5.5% of gate delay errors and 5% of path delay errors.

In this paper, we show how to automatically extract new gate delays for a known new bias voltage. We will demonstrate the adequacy of characterizing a few reference gates instead of all the gates in a library. A new delay for each gate can then be extrapolated from its original delay (at WC or BC) using a simple formula that

factors in the original and new delays (at WC′ or BC′) of the reference gates. Multiple-stage gates and flip-flops need some extension of this concept, and we will address this issue. Output transition time, which is another component of gate timing model, is extracted in a similar manner. Experiments with an industrial 32-nm gate library show that the average errors of extracted gate delays are less than 4.3%. We also create a library of extracted gate delays for a static timing analyzer and examine its accuracy.

The remainder of this paper is organized as follows. Our approach to extracting gate delays is described in the next section; the extraction of output transition times is addressed in Sec. 3. The extension to handle complex gates is addressed in Sec. 4. Experimental results are presented in Sec. 5, and then we draw some conclusions.

2. Extraction of Gate Delays

2.1. *Preliminaries*

Consider the inverter shown in Fig. 2. Its propagation delay d_p is the time between the input voltage V_i reaching $V_{dd}/2$ and the output voltage V_o reaching the same level. We introduce the 50% delay $d_{0.5}$ as the delay from the time when V_i starts to change to the time when V_o becomes $V_{dd}/2$, as illustrated in Fig. 2. If the graph of V_i before and after it reaches $V_{dd}/2$ is rotationally symmetric, then

$$d_{0.5} = d_p + \frac{t_i}{2}\,, \tag{1}$$

where t_i is the transition time of V_i, i.e., input transition time.

In the context of static timing analysis, d_p is usually modeled as a function of t_i and the load (capacitance) C_L. This function may be an empirical expression such as a k-factor equation or a 2D lookup table. We assume, but do not require, that a lookup table is used.

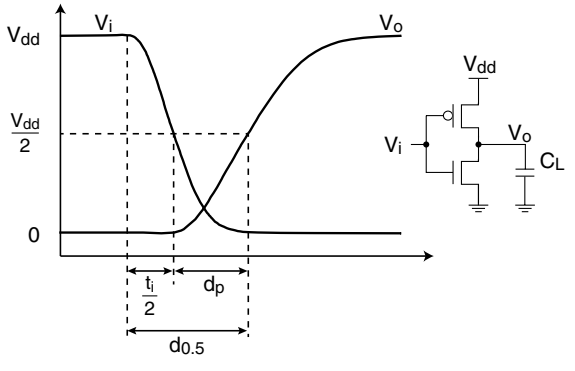


Fig. 2. Timing parameters.

2.2. *Modeling the 50% delay*

Consider the inverter shown in Fig. 2 again. We want to derive an analytic expression for $d_{0.5}$, from which we will be able to estimate the 50% delay $d'_{0.5}$ of a body-biased circuit. For a falling input V_i, the load C_L is charged from a voltage of 0 to $V_{dd}/2$ by the saturation current through the pMOS transistor, which can be approximated as follows:

$$I_d(t) \approx kW(V_{gs}(t) - V_t)\,, \tag{2}$$

where k is a constant, W is the channel width, and V_t is a threshold voltage. The gate-to-source voltage V_{gs} is given by

$$V_{gs}(t) = \begin{cases} -\dfrac{V_{dd}}{t_i}t & 0 \le t < t_i\,, \\ -V_{dd} & t_i \le t\,, \end{cases} \tag{3}$$

again assuming that V_i is a falling ramp which starts at $t = 0$.

We require that

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

where t_1 is the time when the pMOS is turned on: $t_1 = -V_t t_i/V_{dd}$ from $-t_1 V_{dd}/t_i = V_t$. Substituting (3) into (2) and then (2) into (4), and going on to solve (4) for $d_{0.5}$, we obtain,

$$d_{0.5} = \begin{cases} -\dfrac{V_t\,t_i}{V_{dd}} + \sqrt{\dfrac{t_i}{|k|\,(W/C_L)}} & d_{0.5} < t_i\,, \\ \dfrac{t_i(V_{dd} - V_t)}{2V_{dd}} + \dfrac{V_{dd}}{2|k|(W/C_L)(V_{dd} + V_t)} & t_i \le d_{0.5}\,. \end{cases} \tag{5}$$

We compared results from this analytical model with equivalent data obtained by SPICE simulation. Figure 3 shows that there is a good match between the two, in particular when t_i is smaller than 100 ps. Larger values of t_i correspond to larger short-circuit currents, and thus higher value of $d_{0.5}$ must be expected, since in this case it takes longer to charge C_L; our model (5) does not allow for this effect, which increases the error, but values of t_i in the hundreds do not occur very often in practical circuits.

2.3. *Extraction of the 50% delay of body-biased circuits*

Let us assume that body bias is applied to the pMOS transistor in Fig. 2, and we want to derive $d'_{0.5}$, which is the body-biased 50% delay. The only parameter in our

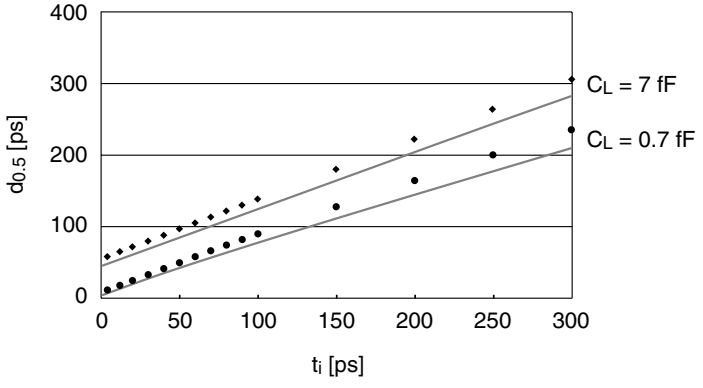


Fig. 3. Results from our analytical model (curve) and SPICE simulation (dots), for a minimum-size inverter loaded with the values of C_L shown.

analytical model (5) that is affected by body bias is the new threshold voltage V_t'; thus $d_{0.5}'$ is modeled in the same way, except that V_t is replaced by V_t'. The question we will now address is how $d_{0.5}'$ can be derived from $d_{0.5}$, which can itself be obtained from the library value d_p (see (1)). There are two cases to consider:

2.3.1. *When* $d_{0.5}' < t_i$

$$d_{0.5}' = -\frac{V_t'\, t_i}{V_{dd}} + \sqrt{\frac{t_i}{|k|\,(W/C_L)}} \tag{6a}$$

$$= \left(-\frac{V_t'\, t_i}{V_{dd}} + \sqrt{\frac{t_i}{|k|\,(W_{\text{ref}}/C_L)}}\right) + \left(-\frac{V_t\, t_i}{V_{dd}} + \sqrt{\frac{t_i}{|k|\,(W/C_L)}}\right)$$

$$-\left(-\frac{V_t\, t_i}{V_{dd}} + \sqrt{\frac{t_i}{|k|\,(W_{\text{ref}}/C_L)}}\right) \tag{6b}$$

$$= d_{0.5,\text{ref}}' + d_{0.5} - d_{0.5,\text{ref}}\,, \tag{6c}$$

where the subscript ref indicates a reference gate such as an inverter. Expression (6c) implies that, once $d_{0.5,\text{ref}}'$ has been obtained for a reference gate, $d_{0.5}'$ can be obtained for any gate, since the other two quantities, $d_{0.5}$ and $d_{0.5,\text{ref}}$, are available from a library. Note that we have assumed that V_t' is independent of the gate type; this assumption is refined in Sec. 4 for the gates of multiple stacks or multiple stages and flip-flops.

2.3.2. *When* $t_i \le d'_{0.5}$

$$d'_{0.5} = \frac{t_i(V_{dd} - V'_t)}{2V_{dd}} + \frac{V_{dd}}{2|k|(W/C_L)(V_{dd} + V'_t)} \tag{7a}$$

$$= \left(\frac{t_i(V_{dd} - V'_t)}{2V_{dd}} + \frac{V_{dd}}{2|k|(W_{\text{ref}}/C_L)(V_{dd} + V'_t)}\right)$$

$$+ \frac{V_{dd} + V_t}{V_{dd} + V'_t}\left(\frac{t_i(V_{dd} - V_t)}{2V_{dd}} + \frac{V_{dd}}{2|k|(W/C_L)(V_{dd} + V_t)}\right)$$

$$- \frac{V_{dd} + V_t}{V_{dd} + V'_t}\left(\frac{t_i(V_{dd} - V_t)}{2V_{dd}} + \frac{V_{dd}}{2|k|(W_{\text{ref}}/C_L)(V_{dd} + V_t)}\right) \tag{7b}$$

$$= d'_{0.5,\text{ref}} + \frac{V_{dd} + V_t}{V_{dd} + V'_t}(d_{0.5} - d_{0.5,\text{ref}})\,. \tag{7c}$$

The implication of (7c) is again the same as that of (6c).

Main idea of delay extraction is to borrow each term of the analytic model from zero and body biased reference gates to compose the analytic model of body biased gate delay. In case of $d'_{0.5} < t_i$, first term of (6a) is equal to that of body biased reference gate because this term is independent on gate type. Similarly, second term is equal to that of zero biased gate because this term is independent on body voltage. These two terms are obtained by simple calculation with zero and body biased reference gates and zero biased gate. In case of $t_i \le d'_{0.5}$, each term of (7a) is obtained with little modification; because second term is also dependent on body voltage, $(V_{dd} + V_t)/(V_{dd} + V'_t)$ is multiplied to delay models of zero biased gates.

Once $d'_{0.5}$ has been obtained from the two key equations, (6c) and (7c), the propagation delay with body bias, denoted by d'_p, can readily be obtained from (1). To build a new 2D lookup table of propagation delay, the computation is repeated while we change the values of t_i and C_L.

3. Extraction of Output Transition Times

3.1. *Model*

The model of 50% delay $d_{0.5}$ introduced in Sec. 2.2 can be generalized for use in output transition time t_o. In recent technology, in which transition time is practically very short, t_o is often defined to be the interval between the output voltage V_o reaching 30% and 70% of V_{dd},[9] i.e.,

$$t_o = d_{0.7} - d_{0.3}\,, \tag{8}$$

for rising output, where d_x is a generalized definition of $d_{0.5}$. We require similar expression to (4):

$$\int_{t_1}^{d_x} I_d(t)dt = C_L x V_{dd}\,. \tag{9}$$

It is easy to show that (5) is generalized to

$$d_x = \begin{cases} -\dfrac{V_t\, t_i}{V_{dd}} + \sqrt{\dfrac{2xt_i}{|k|\,(W/C_L)}} & d_x < t_i\,, \\ \dfrac{t_i(V_{dd}-V_t)}{2V_{dd}} + \dfrac{xV_{dd}}{|k|(W/C_L)(V_{dd}+V_t)} & t_i \le d_x\,. \end{cases} \tag{10}$$

3.2. *Extraction*

The body-biased output transition time, denoted by t'_o, requires the computation of $d'_{0.7}$ and $d'_{0.3}$, i.e., $t'_o = d'_{0.7} - d'_{0.3}$. This can be done in three different cases by referring to (10) because $d_{0.7}$ and $d_{0.3}$ has multiple cases according to t_i.

3.2.1. *When* $d'_{0.7} < t_i$

$$d'_{0.7} - d'_{0.3} = -\left(\frac{V'_t\, t_i}{V_{dd}} + \sqrt{\frac{7t_i}{5|k|\,(W/C_L)}}\right) - \left(-\frac{V'_t\, t_i}{V_{dd}} + \sqrt{\frac{3t_i}{5|k|\,(W/C_L)}}\right) \tag{11a}$$

$$= \sqrt{\frac{7t_i}{5|k|\,(W/C_L)}} - \sqrt{\frac{3t_i}{5|k|\,(W/C_L)}} \tag{11b}$$

$$= d_{0.7} - d_{0.3}\,. \tag{11c}$$

Output transition time remains unchanged in this case.

3.2.2. *When* $d'_{0.3} < t_i \le d'_{0.7}$

$$d'_{0.7} - d'_{0.3} = \left(d'_{0.7,\mathrm{ref}} + \frac{V_{dd}+V_t}{V_{dd}+V'_t}(d_{0.7} - d_{0.7,\mathrm{ref}})\right) - \left(d'_{0.3,\mathrm{ref}} + d_{0.3} - d_{0.3,\mathrm{ref}}\right) \tag{12a}$$

$$= d'_{0.7,\mathrm{ref}} - d'_{0.3,\mathrm{ref}} + \frac{V_{dd}+V_t}{V_{dd}+V'_t}(d_{0.7} - d_{0.7,\mathrm{ref}}) - (d_{0.3} - d_{0.3,\mathrm{ref}})\,. \tag{12b}$$

A similar method of Sec. 2.3 can be applied to obtain $d'_{0.7}$ and $d'_{0.3}$. To obtain $d'_{0.7,\mathrm{ref}}$ and $d'_{0.3,\mathrm{ref}}$, we make the following approximations:

$$d'_{0.7,\mathrm{ref}} = d'_{0.5,\mathrm{ref}} + \frac{t_o}{2}\,, \tag{13a}$$

$$d'_{0.3,\mathrm{ref}} = d'_{0.5,\mathrm{ref}} - \frac{t_o}{2}\,. \tag{13b}$$

Recall the definition of t_o in (8).

3.2.3. *When* $t_i \le d'_{0.3}$

$$d'_{0.7} - d'_{0.3} = \left(\frac{t_i(V_{dd} - V'_t)}{2V_{dd}} + \frac{7V_{dd}}{10|k|(W_{\mathrm{ref}}/C_L)(V_{dd} + V'_t)}\right) - \left(\frac{t_i(V_{dd} - V'_t)}{2V_{dd}} + \frac{3V_{dd}}{10|k|(W_{\mathrm{ref}}/C_L)(V_{dd} + V'_t)}\right) \tag{14a}$$

$$= \frac{4V_{dd}}{10|k|(W_{\mathrm{ref}}/C_L)(V_{dd} + V'_t)} \tag{14b}$$

$$= \frac{V_{dd} + V_t}{V_{dd} + V'_t}(d_{0.7} - d_{0.3})\,. \tag{14c}$$

The overall computations of $d'_{0.7} - d'_{0.3}$ through three key equations, (11c), (12b), and (14c) are performed for each t_i and C_L to build a new 2D lookup table for t'_o.

4. Refinements to Our Model

4.1. *Multiple stacks*

In Sec. 2.3, we ignored the dependence of the threshold voltages, both V_t and V'_t, on gate type. Consider the NOR2 gate in Fig. 4(b). When M_2 charges the load capacitance, its effective threshold voltage is different from that of the pMOS transistor in the basic inverter shown in Fig. 4(a), due to the nonzero voltage drop across

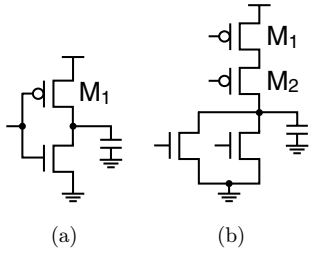


Fig. 4. (a) Inverter and (b) NOR2 gate.

M_1, which is turned on; the value of V_{gs} is not exactly predicted by (3), for the same reason. And when M_1 charges the load capacitance, it also charges the intrinsic capacitance between M_1 and M_2; which causes another deviation from our model.

Our model can be refined to take account of these effects by characterizing $d'_{0.5,\text{ref}}$ for more than one reference gate. Thus we would have one value of $d'_{0.5,\text{ref}}$ for gates that contain a single pMOS stack, such as an inverter or NAND2 gate, two values of $d'_{0.5,\text{ref}}$ (one for M_1 and another for M_2 in Fig. 4(b)) for gates, such as a NOR2 gate, in which two pMOS transistors are stacked, and so on. Since the maximum number of stacked transistors is 3 or 4 in the gate in a typical library, 6 to 10 values of $d'_{0.5,\text{ref}}$ need to be determined.

4.2. *Multiple stages*

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

$$d_p = d_{p,1} + d_{p,2}\,, \tag{15}$$

which respectively correspond to the propagation delay of its internal NAND and INV gates. Note that these internal propagation delays are unknown since only d_p is available from a library. We will now consider what happens when body bias is applied to the pMOS transistors; we assume that effect of body bias on pMOS is negligible when load capacitance is discharged.

4.2.1. *Falling ramp applied to V_i*

Application of body bias affects $d_{p,1}$ but not $d_{p,2}$ (if we ignore the effect of the change in signal transition time of V_{o1} on $d_{p,2}$). Thus,

$$d'_p = d'_{p,1} + d_{p,2} \tag{16a}$$

$$= d'_{0.5,1} - \frac{t_i}{2} + d_{p,2}\,. \tag{16b}$$

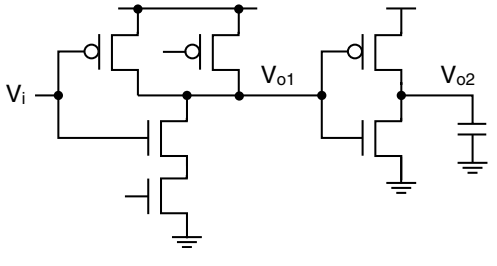


Fig. 5. AND2 gate.

We can approximate $d_{p,2}$ by a simple linear model:

$$d_{p,2} = \eta\, C_L\,, \tag{17}$$

where η is a constant of proportionality that takes a value that depends on t_i. Thus (16) becomes

$$d'_p = d'_{0.5,1} - \frac{t_i}{2} + \eta\, C_L\,. \tag{18}$$

Note that $d'_{0.5,1}$ can be obtained by using either (6c) or (7c); these equations require $d_{0.5,1}$, which can be expressed as follows:

$$d_{0.5,1} = d_p - \eta\, C_L + \frac{t_i}{2}\,. \tag{19}$$

The remaining task in determining d'_p is to obtain η for each gate which has multiple stages. This is a simple calculation. Since $d_p = d_{p,1} + \eta\, C_L$, the two propagation delays d_{pA} and d_{pB} can be read from a delay table for the respective load capacitances C_{LA} and C_{LB}. Provided that $d_{p,1}$ is invariant in the resulting expressions for d_{pA} and d_{pB}, we can take the difference between these expressions and solve for η:

$$\eta = \frac{d_{pA} - d_{pB}}{C_{LA} - C_{LB}}\,. \tag{20}$$

By repeating this process for different values of t_i, we obtain an array of η values.

4.2.2. *Rising ramp applied to V_i*

In this case the body bias affects the delay of INV:

$$\begin{aligned} d'_p &= d_{p,1} + d'_{p,2} && \text{(21a)}\\ &= (d_p - d_{p,2}) + d'_{p,2} && \text{(21b)}\\ &= (d_p - \eta\, C_L) + d'_{0.5,2} - \frac{t_{o1}}{2}\,, && \text{(21c)} \end{aligned}$$

where t_{o1} is the signal transition time of V_{o1}. In a similar way to that in which we obtained $d'_{0.5,1}$ in the previous section, we can now obtain $d'_{0.5,2}$ using (6c) or (7c), which in turn requires $d_{0.5,2} = d_{p,2} + t_{o1}/2 = \eta\, C_L + t_{o1}/2$. The only unknown that is left is t_{o1}, which can be approximated from a table of output transition times of a standalone NAND gate (in case of AND gate shown in Fig. 5) indexed by t_i and a load capacitance equal to the input capacitance of an inverter of minimum size.

4.3. *Flip-flops*

Consider the rising-edge-triggered D flip-flop shown in Fig. 6. If input data D passes through the first transmission gate after the clock signal C has gone high, then D is

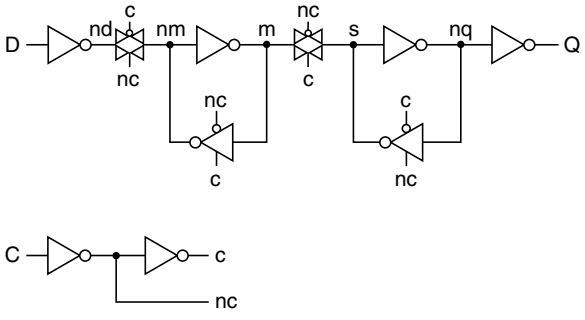


Fig. 6. *D* flip-flop.

captured by the master latch, causing a hold time violation. Thus, the hold time can be expressed as follows:

$$t_h = d_{C-nc} - d_{D-nd}\,, \tag{22}$$

where d_{i-j} is the delay from i to j. Now consider what happens when body bias is applied to all the pMOS transistors in the flip-flop. We want to find the new hold time t'_h. It is clear that d_{C-nc} is invariant under body bias, since the associated inverter is triggered by a rising clock. If D is rising, d_{D-nd} is also invariant and thus $t'_h = t_h$. If D is falling, then $t'_h = d_{C-nc} - d'_{D-nd} = d_{C-nc} - (d'_{0.5,D-nd} - t_i/2)$. We can see from (6c) and (7c) that $d_{0.5,D-nd}$ is required to obtain $d'_{0.5,D-nd}$; but $d_{0.5,D-nd}$ can be calculated from (22) and (1), provided that d_{C-nc} is available. Therefore, we only need to know d_{C-nc}, which is an array of value for different signal transition times of D, to obtain t'_h.

If the clock-to-Q delay exceeds a typical value measured when D arrives in time, by some specified percentage, then a setup time violation is considered to have occurred; this is confirmed if m fails to stabilize before the second transmission gate opens, so that

$$t_s = d_{D-m} - d_{C-nc} \tag{23a}$$

$$= d_{D-nm} + d_{nm-m} - d_{C-nc}\,. \tag{23b}$$

To obtain t'_s for rising D, we need to find d'_{nm-m}, since the other two quantities are invariant. We characterize d_{D-nm}, which allows us to determine d_{nm-m}; we also characterize t_{nm}, and so d'_{nm-m} can now be obtained using the method of Sec. 2.3. If D is falling, then $t'_s = d'_{D-nm} + d_{nm-m} - d_{C-nc}$. We characterize d_{D-nm} again, but this time for a falling input; this also allows us to calculate the new value of d_{nm-m}, and thus we readily obtain d_{D-nm}. In summary, the three quantities d_{D-nm}, for both rising and falling input, and t_{nm}, which are all arrays of values, are characterized to determine t'_s.

The clock-to-Q delay can be approximated by

$$d_{cq} = d_{C-nc} + d_{s-Q} \tag{24a}$$
$$= d_{C-nc} + d_{s-nq} + d_{nq-Q}\,. \tag{24b}$$

We first calculate d_{s-Q} (d_{C-nc} is now available and d_{cq} can be obtained from a library); d_{nq-Q} is then approximated by a linear model (17); d_{s-nq} is now calculated and stored. To obtain d'_{cq} for falling Q, we only need d'_{s-nq}, which can be calculated once we know t_s; and it turns out that t_s can be approximated by t_{nc}, which we characterize. For rising Q, we need d'_{nq-Q}, which can be obtained if we know t_{nq}, which is approximated by a table of output transition times for a standalone inverter, indexed by t_s with a load capacitance equal to the input capacitance of another inverter, of minimum size.

5. Experiments

Experiments were performed using an industrial 0.9 V 32-nm gate library, in which the propagation delay and output transition time of each gate, respectively, is given as a 7×7 table indexed by input transition time and load capacitance. The setup- and hold-times of flip-flops are modeled by 3×3 tables indexed by input and clock transition times.

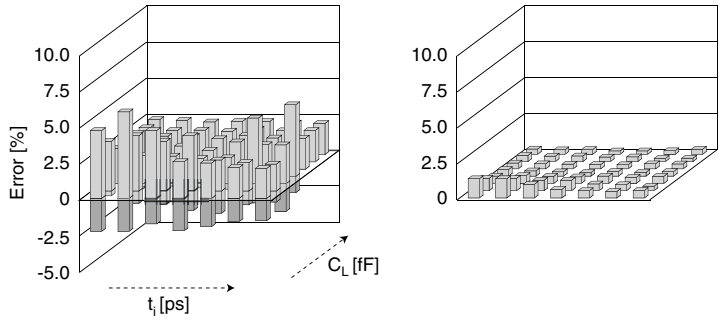
We considered the application of a forward body bias of 0.35 V to the pMOS transistors. This is the maximum bias voltage allowed in the technology we based on, and causes the largest error in our method; the error would be less if a lower bias voltage were applied, as we will discuss in Sec. 5.1. We generated new delay tables analytically to allow for body bias, and compared them with delay tables obtained by characterization using SPICE. For the proposed method, total 12 tables of $d_{0.5,\mathrm{ref}}$ were created by SPICE simulation, using INV, NOR2, and NOR3 as reference gates including rising and falling input cases; four arrays giving d_{C-nc}, two d_{D-nm}, and t_{nc} were created from the result of SPICE simulation of a minimum-size D flip-flop.

5.1. *Assessment of extracted gate delay*

5.1.1. *Single-stage gates*

For each of the 58 single-stage gates, a 7×7 table of percentage errors in the estimated delay was constructed. The maximum and average errors in these tables are shown in Fig. 7: the maximum error ranges from -2.3% to 6.1%; the average errors are all below 1.0% (positive errors mean underestimation and negative errors mean overestimation, respectively).

Larger errors occur at gates such as a NAND4. Our estimate is based on $d_{0.5,\mathrm{ref}}$ and $d'_{0.5,\mathrm{ref}}$, which are obtained using an inverter as a reference gate. Consider the



(a) (b)

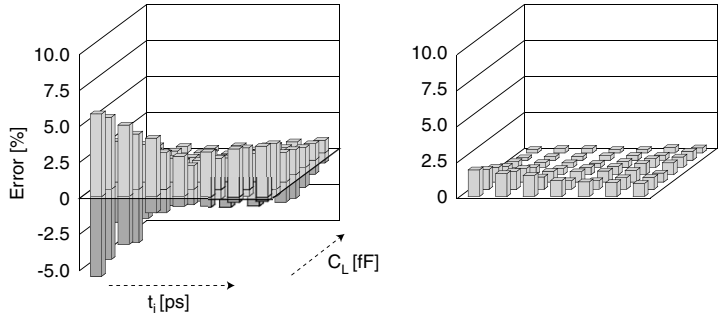
Fig. 7. Single-stage gates: (a) maximum and (b) average errors.

application of a falling transition to one of the pMOS transistors, which is paired with the lowest nMOS while the other inputs are tied to 1. The load capacitance starts to be charged, but so are the intrinsic capacitances in the pull-down network. This causes an error because it does not happen in a reference inverter; the error is especially marked when the load capacitance is sufficiently small to become comparable to the intrinsic capacitances.

5.1.2. *Multiple-stage gates*

The errors in the 37 multiple-stage gates are shown in Fig. 8. The maximum error ranges from -6% to 6.0% and the average errors are all below 2.0%, which are comparable to the error ranges in single-stage gates without negative errors.

The main source of error is (17), which treats the delay in the second stage as linear. As C_L decreases, the error increases because the constant intrinsic delay in the second stage, which is not taken into account in (17), becomes more comparable to



(a) (b)

Fig. 8. Multiple-stage gates: (a) maximum and (b) average errors.

the value of (17); in other words, delay in second stage is underestimated. As t_i decreases, this error becomes even more significant because the delay in the first stage becomes relatively shorter. For example, when falling ramp is applied to V_i, the delay value is underestimated; estimated delay in first stage decreases more than characterized one because zero biased delay in first stage is overestimated from (19). Similarly, when rising ramp is applied to V_i, the delay value is overestimated due to delay in second stage, which less decreases. This is clearly shown in Fig. 8. Fortunately, these extreme cases rarely happen in practical circuits, even though they are included in delay tables.

5.1.3. *Flip-flops*

We tested 25 D flip-flops from the library, of different sizes, with and without set or reset pins, and triggered by clock signals of different polarity. The errors in hold time range from 0.07 ps to 8.8 ps, and the setup time errors are between 0.04 and 11.9 ps. Both hold- and setup-times are functions of input and clock transition times; the maximum values of these transition times in the library are respectively 760 and 380 ps. If we ignore these extremes, the largest errors in hold- and setup-times are respectively reduced to 2.2 and 4.2 ps. The errors in clock-to-Q delay range from 0.08 and 9.1 ps.

Even though only four arrays of timing parameters were computed in advance for a minimum-size rising-edge-triggered D flip-flop, the results for all flip-flops are of reasonable accuracy. This is because d_{C-nc}, d_{D-nm}, and t_{nc} (see Fig. 6) vary very little across the flip-flops we tested.

5.1.4. *Effect of bias voltage on accuracy*

We have assumed so far that a forward body bias of 0.35 V is applied to all pMOS transistors. As this voltage is reduced, the errors become smaller; e.g., when the bias voltage is 0.1 V, the maximum error for all gates is reduced to 1.7%, which can be compared to 6.1% for bias voltage of 0.35 V. We can explain this by recollecting that the key assumption that we made in deriving $d'_{0.5}$ in Sec. 2.3 is that the threshold voltage with body bias, V'_t, can be obtained from values from a reference gate. The validity of this assumption improves when the bias voltage is reduced, and V'_t is closer to the original threshold voltage.

5.2. *Assessment of extracted output transition time*

The accuracy of extracted output transition times depends on the input transition time, for which the extraction is performed. When input changes very slowly, the extracted output transition time is less accurate because the current model that we assume in (2) only contains the saturation current. The two largest input transition

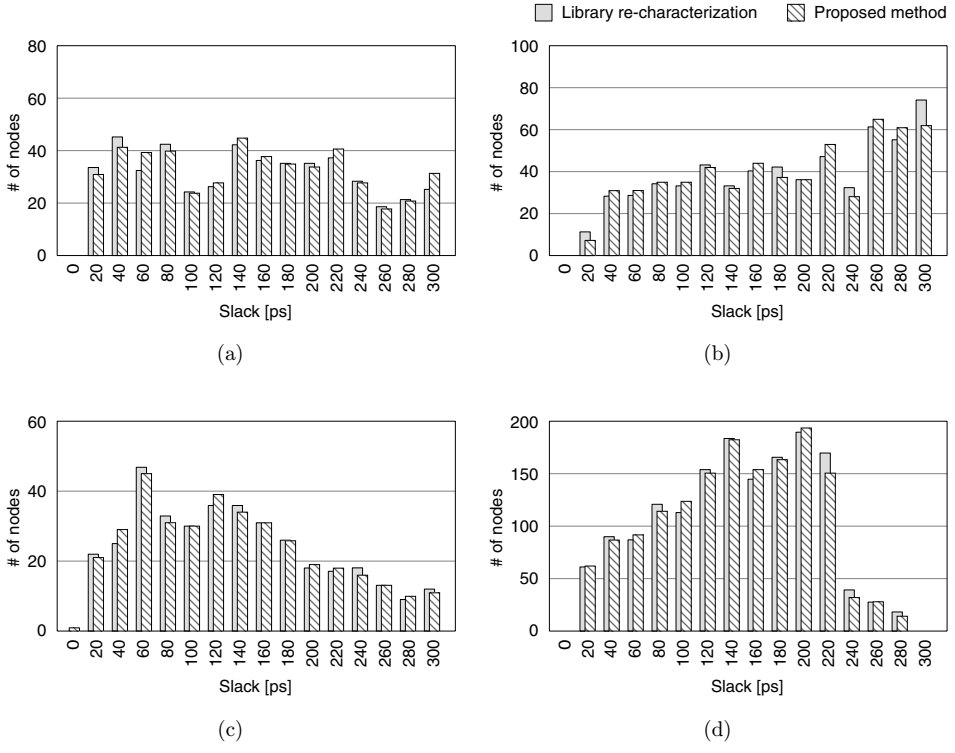


Fig. 9. Comparison of slack histograms for circuit (a) s9234, (b) s13207, (c) s1423, and (d) s5378 between library re-characterization and proposed method.

times in 2D lookup table are 500 and 760 ps. The maximum percentage error is relatively large, 25%, in these cases, but these input transition times are extreme and are not likely to occur in practical circuits. For input transition times smaller than 500 ps, the largest error in extracted output transition times is 5 ps, which is acceptable.

5.3. *Application to static timing analysis*

We have shown that our technique can generate reasonably accurate gate delays. To see how well these delays work within a static timing analysis, we prepared two libraries: one containing generated gate delays and output transition times, and the other compiled by re-characterizing gate delays using SPICE.

Slack histograms obtained by applying the two libraries to example test circuits match very well as shown in Fig. 9. Figures 7 and 8 indicate that gate delays are sometimes under- and sometimes over-estimated. Path delay, however, is kept accurate because gate delay errors tend to be canceled out.

6. Conclusion

We have proposed a method of extracting gate delays for timing analysis of body-biased circuits. Our key observation is that the new delay with body bias of a gate can be obtained from its old delay without body bias, together with old and new delays of a reference gate. A similar idea can be applied to extract the new output transition times. The accuracy of the resulting delays has been assessed in the context of a 32-nm library. The applicability of this approach to static timing analysis has been demonstrated.

Acknowledgments

The authors would like to thank Dr. Hyung-Ock Kim, Dr. Jun Seomun, Mr. Jaehan Jeon, Dr. Jung Yoon Choi, Dr. Hyo-Sig Won, and Dr. Kee Sup Kim from Samsung Electronics for their help and support in this work.

References

1. B. Choi and Y. Shin, Lookup table-based adaptive body biasing of multiple macros, *Proc. Int. Symp. Quality Electronic Design*, March 2007, pp. 533–538.
2. J. Tschanz, J. Kao, S. Narendra, R. Nair, D. Antoniadis, A. Chandrakasan and V. De, Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage, *IEEE J. Solid-State Circuits* **37** (2002) 1396–1402.
3. T. Chen and S. Naffziger, Comparison of adaptive body bias (ABB) and adaptive supply voltage (ASV) for improving delay and leakage under the presence of process variation, *IEEE Trans. VLSI Syst.* **11** (2003) 888–899.
4. S. Narendra, A. Keshavarzi, B. Bloechel, S. Borkar and V. De, Forward body bias for microprocessors in 130-nm technology generation and beyond, *IEEE J. Solid-State Circuits* **38** (2003) 696–701.
5. S. Kulkarni, D. Sylvester and D. Blaauw, Design-time optimization of post-silicon tuned circuits using adaptive body bias, *IEEE Trans. Comput.-Aided Design* **27** (2008) 481–494.
6. S. Narendra and A. Chandrakasan (eds.), *Leakage in Nanometer CMOS Technologies* (Springer, 2005).
7. M. Meijer and J. P. de Gyvez, Body bias driven design synthesis for optimum performance per area, *Proc. Int. Symp. Quality Electronic Design*, March 2010, pp. 472–477.
8. K. Shinkai, M. Hashimoto, A. Kurokawa and T. Onoye, A gate delay model focusing on current fluctuation over wide-range of process and environmental variability, *Proc. Int. Conf. Computer Aided Design*, November 2006, pp. 47–53.
9. S. Hu, C. Alpert, J. Hu, S. Karandikar, Z. Li, W. Shi and C. Sze, Fast algorithms for slew-constrained minimum cost buffering, *IEEE Trans. Comput.-Aided Design* **26** (2007) 2009–2022.