Chip Optimization Through STI Stress-Aware Placement Perturbations and Fill Insertion

Andrew B. Kahng  Puneet Sharma  Rasit O. Topaloglu

Abstract—Starting at the 65nm node, stress engineering to improve performance of transistors has been a major industry focus. An intrinsic stress source – shallow trench isolation (STI) – has not been fully utilized up to now for circuit performance improvement. In this paper, we present a new methodology that combines detailed placement and active-layer fill insertion to exploit STI stress for performance improvement. We conduct process simulation of a production 65nm STI technology to generate mobility and delay impact models for STI stress. We then utilize these models to perform STI stress-aware delay analysis of critical paths using SPICE. We present our timing-driven optimization of STI stress in standard cell designs, using detailed placement perturbation and active-layer fill insertion to improve CMOS performance. We assess the proposed analysis and optimization on small designs implemented with a 65nm production cell library and a standard simulation, place and route flow. Our stress-aware timing analysis improves clock frequency by 4.68% to 6.31% over traditional worst-case analysis, and our optimization improves clock frequency by 2.44% to 5.26%. The frequency improvement through exploitation of STI stress comes at practically zero cost in terms of design area and wirelength.

I. INTRODUCTION

At the 65nm process node and beyond, it is evident that stress- and strain-based techniques for mobility improvement will dominate traditional geometric scaling to maintain Moore’s Law trajectories for device performance. Enabling progress has been made in the manufacturing process and TCAD (modeling and simulation) to support stress. However, stress has not yet been exploited by layout optimizations to improve design performance. In this work, we present a new methodology that combines detailed placement and active-layer fill insertion to exploit shallow-trench isolation (STI) stress for IC performance improvement. Our methodology begins with process simulation of a production 65nm STI technology, from which we generate mobility and delay impact models for STI stress. We develop STI stress-aware SPICE modeling and simulation of critical paths, and finally perform timing-driven optimization of STI stress in standard cell designs, using detailed placement perturbation to optimize PMOS performance and active-layer fill insertion to optimize NMOS performance.

Stress Modulation Techniques in the Process.

In 65nm processes, a number of stress modulation techniques have been introduced:

- SiGe stress from underneath the channel
- Embedded SiGe from the source and drain
- Stress liner
- Stress memorization
- Hybrid orientation

Early stress modulation methods employed a Silicon-Germanium (SiGe) layer underneath the channel, which improves the channel mobility. More recently, embedded SiGe (e-SiGe) [10], [11], [13] has been used in the source and drain regions to exert stress along the channel, and improve PMOS speed. The stress liner technique involves deposition of stressed liners over the transistors on top of the polysilicon. Single stress liners may be used to cover the entire wafer with compressive or tensile liners. Alternatively, dual stress liners [16], [15] may be used to cover the NMOS devices with a tensile liner and PMOS devices...
with a compressive one. Stress memorization technique [17], [18], used typically to improve NMOS speed, relies on plastic deformation of certain materials due to a process step and the consequent memorization of the applied stress in the channel. Finally, in the hybrid orientation technique [14], [12], crystal orientations are used to enhance NMOS and PMOS speeds separately.

STI stress is the stress that is exerted by STI wells on device active regions and is generally compressive in nature. Irrespective of the use of stress modulation techniques in the process, STI stress is not negligible and its magnitude depends on the sizes of the STI wells and the active regions for a given process. In this paper, we present a technique that modulates stress at timing critical devices to improve circuit delay by altering the STI widths (STIW) adjacent to the devices.

**Stress Modeling Techniques.** In the area of stress modeling and characterization, Rueda et al. [7] have provided general models for stress. Gallon et al. [22] have specifically analyzed the stress induced by STI. Bradley et al. [23] have characterized the piezo-resistance of CMOS transistors. Sheu et al. [28] have modeled well edge proximity effect on MOSFETs, and [21] have modeled the effect of mechanical stress on dopant diffusion. Su et al. [24] have proposed a scalable model for layout dependence of stress. Miyamoto et al. [29] have provided a layout dependent stress analysis of STI. Recently, Tsuno et al. [30] have shown 65nm silicon data showing that STI width stress effect can impact drive current by up to 10%. However, no models or optimization methodology is presented.

With respect to the STI process, several optimizations have been developed to reduce the STI stress, but they typically fail to completely eliminate layout-dependent stress impacts. Elbel et al. [19] have proposed an STI process flow based on selective oxide deposition. Lee et al. [20] have proposed an optimization for densification of the STI fill oxide to reduce the stress. Miyamoto et al. [29] also have proposed process innovations to reduce the active-area layout dependence of MOSFET characteristics. Looking forward, the introduction of e-SiGe for certain 65nm nodes in the source and drain may reduce the stress variation due to STIW for PMOS, but NMOS performance can still be actively improved by exploiting the STI width effect. If these STI stress controlling techniques are used, we expect the improvements demonstrated in this paper to reduce but remain non-negligible.

Stress TCAD simulations have been conducted by Moroz et al. in [25] and [27], and by Smith in [26]. The work of Moroz et al. is significant for indicating possible ways to enhance performance using STI stress; however, no circuit-level optimizations are explained. With respect to the current body of knowledge: (1) models are still needed to relate stress due to the STI width effect to transistor mobilities, and (2) there is still a lack of available stress optimization methods. A fundamental research goal is to develop novel and efficient simulation, modeling, analysis, and optimization methods to support next-generation stress-aware EDA technology. Our work strives to enable this.

**Our Focus: Exploitation of STI Width Effect.** STI is an important and well-studied stress source that has not been fully exploited until now for design quality improvement. STI usually exerts a compressive stress along the channel (i.e., the current flow direction), which improves PMOS device mobility. The opposite type of stress, tensile stress, degrades the PMOS performance in this direction. NMOS is in general complementary to PMOS in terms of how it is affected by stress, and its mobility degrades because of STI stress.

Device mobility increase corresponds to speed increase. Hence, it is possible to utilize STI, which is used to separate NMOS and PMOS regions, to improve performance. Table I shows the impact of STIW on rise and fall delays (averaged over all timing arcs) of several 65nm standard cells using the models developed in this paper. Figure 1 illustrates the change in STI width for the INVD0 when the inter-cell spacing is increased from 0µm (i.e., abutting neighbors) to 5µm. Impact of placement on STI width and consequently on rise (R-Delay) and fall (F-Delay) delays for a few cells are provided. For each cell in the table, three instances of it are placed with different spacings between them, and the delay of the center instance is reported. In Table I, **Spacing** is the spacing between cells and **PMOS STIW** (NMOS STIW) and **PMOS STIW** (PMOS STIW) are the STI widths next to the left and right sides of positive active (negative active) regions of the center cell. It is possible to both speed up and slow down cells by controlling the STIW and, thereby, the stress that is applied to a cell. In particular, larger STI width will generate more stress in neighboring transistors. In this paper, we propose placement perturbation and the insertion of active-layer fills to control the STI width in a performance-driven manner.

The proposed active-layer fill insertion and placement perturbation do not require additional process steps or add complications to resolution enhancement techniques. Active-layer fill insertion is a standard process step that is performed in all designs to control active-layer density. Placement perturbation yields a new valid placement. We ensure that the design is design rule correct after we perform these two steps.

**Organization of the Paper.** The remainder of this paper is organized as follows. In the next section, we describe our STI width-induced stress models that we have developed. After a brief introduction to stress, we review the process steps we have simulated using TCAD tools, and our proposed stress models. In
TABLE I
IMPACT OF STI WIDTH ON PERFORMANCE OF SEVERAL STANDARD CELLS.

<table>
<thead>
<tr>
<th>Cell</th>
<th>Spacing (nm)</th>
<th>PMOS STIW&lt;sub&gt;L&lt;/sub&gt; (nm)</th>
<th>PMOS STIW&lt;sub&gt;R&lt;/sub&gt; (nm)</th>
<th>NMOS STIW&lt;sub&gt;L&lt;/sub&gt; (nm)</th>
<th>NMOS STIW&lt;sub&gt;R&lt;/sub&gt; (nm)</th>
<th>R-Delay (ps)</th>
<th>F-Delay (ps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVD0</td>
<td>0um</td>
<td>140</td>
<td>140</td>
<td>110</td>
<td>110</td>
<td>27.27</td>
<td>21.96</td>
</tr>
<tr>
<td></td>
<td>5um</td>
<td>5140</td>
<td>5140</td>
<td>5110</td>
<td>5110</td>
<td>23.65</td>
<td>23.70</td>
</tr>
<tr>
<td>BUFD0</td>
<td>0um</td>
<td>140</td>
<td>140</td>
<td>125</td>
<td>125</td>
<td>45.56e</td>
<td>46.11</td>
</tr>
<tr>
<td></td>
<td>5um</td>
<td>5140</td>
<td>5140</td>
<td>5125</td>
<td>5125</td>
<td>43.84</td>
<td>43.53</td>
</tr>
<tr>
<td>NR2D0</td>
<td>0um</td>
<td>140</td>
<td>140</td>
<td>110</td>
<td>110</td>
<td>51.12</td>
<td>23.06</td>
</tr>
<tr>
<td></td>
<td>5um</td>
<td>5140</td>
<td>5140</td>
<td>5110</td>
<td>5110</td>
<td>42.77</td>
<td>24.69</td>
</tr>
<tr>
<td>ND2D0</td>
<td>0um</td>
<td>140</td>
<td>140</td>
<td>110</td>
<td>110</td>
<td>42.77</td>
<td>24.69</td>
</tr>
<tr>
<td></td>
<td>5um</td>
<td>5140</td>
<td>5140</td>
<td>5110</td>
<td>5110</td>
<td>25.77</td>
<td>38.81</td>
</tr>
</tbody>
</table>

Based on the understanding in [7], the stress components on a unit cell are as shown in Figure 2. The stress vector \( T_x \) acting normal to \( x \) is given as \( T_x = \sigma_{xx} \hat{x} + \sigma_{xy} \hat{y} + \sigma_{xz} \hat{z} \). The stress tensor is defined by the three stress vectors:

\[
\sigma_{ij} = \begin{bmatrix}
\sigma_{xx} & \tau_{xy} & \tau_{xz} \\
\tau_{yx} & \sigma_{yy} & \tau_{yz} \\
\tau_{zx} & \tau_{zy} & \sigma_{zz}
\end{bmatrix}
\]

In this equation, \( \sigma_{ii} \)'s are stress components normal to the unit cube faces, whereas \( \tau_{ij} \)'s are shear components directed towards \( j \) on the orthogonal face to \( i \). The \( \sigma_{ii} \)'s are used for analyzing the impact of stress. Using the individual stress components, we may convert the stress values to mobility [27].

Fig. 1. Change in STI width on increasing the inter-cell spacing from 0\( \mu m \) to 5\( \mu m \).

Fig. 2. Stress components on a unit volume.

Section III, we present our STI stress-aware timing analysis approach. Section IV describes our timing optimization methodology. In Section V, we present our circuit-level optimization results. Section VI concludes.

II. STI WIDTH STRESS MODELING

The remainder of this section describes a generic STI process flow, along with the STIW models we propose. The STIW parameter is as shown in Figure 3.

A. Process Steps

The process recipe that we use for simulation of STI stress is summarized in Table II. We have simulated the structure up to the gate deposition step using the Synopsys Sentaurus 2005.12 process simulator.\(^1\)

We make the following observations.

- We use a high mesh density, especially between the STI and underneath the channel, to obtain accurate finite-element calculations close to the channel.

\(^1\)When foundry models are used, the exact process steps are not known except for hints provided in the literature or various collateral documentation. Foundries should provide STI width impact models in such a scenario.
work not only models STIW impact, but also builds upon this modeling to improve circuit performance at no area cost.

The STI impact as a function of length of diffusion (LOD) (refer to Figure 3) is already incorporated into the BSIM4 model. Our objective is to isolate and correct for the impact of STIW, in a manner that can be applied on top of existing BSIM4 stress modeling. Using 2D simulations, we have developed the model given in Equation (1)-(4) to capture the STIW effect in the parallel direction (shown in Figure 3). The LOD parameter still appears in the equation, as the STIW impact differs according to the choice of LOD. Also, for purposes of this paper, we do not require or discuss STIW impact modeling in the orthogonal direction (shown in Figure 3), as the STI width effects are blocked in the orthogonal directions by active regions for the type of standard cells we have used. At the end of TCAD simulations, we obtain stress values in Pascals. We then convert the stress values to mobilities using the methodology in [8] and normalize the mobilities. The NMOS equation is given as:

\[
MOB_{LR} = \xi + (1 - (STIW_{LR}/2)^a)/S[A,B]^b \tag{1}
\]

\[
MOB = [MOB_L \cdot MOB_R]^{0.26} \tag{2}
\]

In Equation (2), MOB is the mobility multiplier. Parameters L and R indicate left and right directions with respect to the channel. The equation states that the final mobility multiplier (i.e., MOB) is the product of the mobility multipliers from the left and right directions (i.e., MOB_L and MOB_R). The PMOS equation is given as:

\[
MOB_{LR} = \xi + ((STIW_{LR}/2)^a)/S[A,B]^b \tag{3}
\]

\[
MOB = [MOB_L \cdot MOB_R]^{0.14} \tag{4}
\]

The model and data comparison is shown in Figure 4. In the figure, the x axis is a given data point in the DOE, i.e., a given SA, SB, STIW_L and STIW_R combination, and the y axis is the mobility multiplier. The models provide an average of 7.5% accuracy with respect to the data, along with preserving physical intuition.

The physical intuition is simply based on a relational understanding of how distance to a stress source impacts the stress depending on the type of transistor. As S[A,B] increases in Equation (3), the stress source becomes farther away from the channel. Hence mobility should decrease for PMOS as STI has compressive stress in this technology. This is captured in the equation. Increasing STIW in the Equation (3), should improve the mobility. This also can be seen from the equation. To the contrary, NMOS equation, Equation (1), has these physical terms to yield the

---

**TABLE II**

**STI PROCESS STEPS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Deposit pad oxide</td>
</tr>
<tr>
<td>2</td>
<td>Deposit pad nitride</td>
</tr>
<tr>
<td>3</td>
<td>Deposit photoresist for STI lithography</td>
</tr>
<tr>
<td>4</td>
<td>Anisotropically etch nitride and oxide</td>
</tr>
<tr>
<td>5</td>
<td>Strip photoresist</td>
</tr>
<tr>
<td>6</td>
<td>Directional etch at a rate of 0.01µm/s for 40 seconds at 86° angle</td>
</tr>
<tr>
<td>7</td>
<td>Ramp up temperature to 600°C</td>
</tr>
<tr>
<td>8</td>
<td>Deposit TEOS oxide at a rate of 0.01µm/s for 20 seconds</td>
</tr>
<tr>
<td>9</td>
<td>Deposit trench fill oxide</td>
</tr>
<tr>
<td>10</td>
<td>Temperature ramp up from 600°C to 1000°C at a rate of 50° C/min</td>
</tr>
<tr>
<td>11</td>
<td>Hold temperature for 1 minute at 1000°C</td>
</tr>
<tr>
<td>12</td>
<td>Temperature ramp down from 1000°C to 600°C at a rate of 50° C/min</td>
</tr>
<tr>
<td>13</td>
<td>Diffuse oxide</td>
</tr>
<tr>
<td>14</td>
<td>STI CMP</td>
</tr>
<tr>
<td>15</td>
<td>Etch nitride isotropically at a rate of 0.015µm/s for 15 minutes</td>
</tr>
<tr>
<td>16</td>
<td>Etch oxide isotropically at a rate of 0.02µm/s for 1 minute</td>
</tr>
<tr>
<td>17</td>
<td>Temperature ramp up from 600°C to 800°C at a rate of 40° C/min</td>
</tr>
<tr>
<td>18</td>
<td>Diffuse for 5 minutes using dry O²</td>
</tr>
<tr>
<td>19</td>
<td>Temperature ramp down from 800°C to 600°C at a rate of 40° C/min</td>
</tr>
<tr>
<td>20</td>
<td>Deposit polysilicon gate</td>
</tr>
<tr>
<td>21</td>
<td>Ramp down to room temperature</td>
</tr>
</tbody>
</table>

---

**B. STI Stress Modeling**

The popularly used BSIM SPICE model (revision 4.3 and higher) contains an explicit STI model. However, only the impact of the distance from transistor channel to the STI boundary is modeled. Hence, the dependency on the STI width is not present in the BSIM4 model. Our simulations, as well as simulations and data in the literature, show that STIW impact cannot be neglected. Thus, as noted above, our present

- Temperature cycling (Steps 7 and 19) and densification steps (Steps 10-12) are responsible for the stress build-up. Due to visco-elastic material behavior, materials cannot recover to their original state after stress is withheld. Thermal cycles result in stress due to thermal mismatch between different materials, which have different thermal expansion coefficients. As a result, stress builds up in the STI oxide and this stress remains there even at room temperature at the end of the process. Final stress shows its effects all the way in to the channel of neighboring transistors, in a space-dependent trend, during the lifetime of a chip.

- In Step 14, STI CMP is applied. At the end of this step, the top of the STI is left above the active region on purpose. The basic reason is to avoid defectivity such as delamination of the STI oxide. At the edges of the channel, this step height difference would introduce threshold voltage variations and so-called “width effects.”

---
opposite behavior. The mathematical constants are used to enable a fitting for the particular process technology.

When BSIM models are being generated from test structures, assuming that device modeling test structures with worst-case corners for STI width are present, the slow N - slow P corner would correspond to data from test structures with large STI widths near NMOS modeling devices, and/or test structures with minimum STI widths near PMOS devices. Our normalization step should consider the slow corner conditions as described herein, as well as the presence of STI models in BSIM models. The latter is necessary to eliminate any possibility of the double counting for the channel to STI edge stress, which is already covered by the BSIM models.

In order to enable an accurate normalization, we have set the slow N or P devices to a mobility of 1. Other STIW values result in an increase in the mobility. As multiple $S_A$ and $S_B$ values which can share the same STIW are present in the DOE, there are multiple devices which has a mobility of 1 after normalization. The models are then fit to the normalized results of the process simulation DOE. The corresponding parameters of the model are given in Table III.

### TABLE III

<table>
<thead>
<tr>
<th>Model Parameter Table</th>
<th>$\zeta$</th>
<th>$\alpha$</th>
<th>$\beta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMOS</td>
<td>1.03</td>
<td>0.076</td>
<td>0.48</td>
</tr>
<tr>
<td>PMOS</td>
<td>0.49</td>
<td>0.48</td>
<td>0.57</td>
</tr>
</tbody>
</table>

In Equations (5) and (6), $i$ is an enumeration over active region edges, $w_i$ and $t_i$ are the width of each such edge parallel to the channel for distance from channel to the active edge and STIW, respectively. The parameter $W$ is the width of the channel. Similar width-weighted averaging is used by BSIM 4.3.0 for calculation of stress effects without explicitly using parameters to capture irregular active region shapes.

A related discussion pertains to the choice of active diffusion fill widths. Fringing effects due to STI next to PMOS may degrade NMOS speed. However, we consider the fringing effects to be small because NMOS and PMOS are separated by hundreds of microns. To further limit this issue, the width of the STI near PMOS can be decreased by a few nanometers. Exact value to be used would depend on particular technology and can be obtained through the foundry or by conducting 3D stress simulations.

### STI CMP

Traditionally, active region fills are usually inserted at the tape-out stage to minimize active region density variations. When active region fills need to be utilized for performance, a number of observations needs to be made. A buffer distance can be determined, such that above this distance, the stress width has no further effect on channel. This distance can be selected as $10\mu m$ for the process we have studied for example. It is advisable that such post-fill insertion algorithms use such buffer distances and do not insert fills inside buffer regions near critical gates. Furthermore, these fills are inserted using a window-based scheme to minimize density variations across windows. With optimization, fills will be inserted next to NMOS but not next to PMOS. Hence, there will be approximately 50% active region fill in an optimized window. If the density is lower and needs to be increased, then users can start to insert fills near non-critical gates as well. This approach will have negligible impact on timing closure.

### Impact of Stress Liner

To evaluate the impact of stress liners on STI stress, we have used a similar simulation setup with the parameters given in Table IV. A nitride liner is shown in Figure 5. The nitride liner height and intrinsic stress are varied to observe the influence on STI stress. The parameters are shown in Figure 3. Combinations of all parameters are simulated individually. We have observed that, in the given parameter range, the addition of a $1GPa$ $0.1\mu m$ thick stressed nitride layer increases the impact of STI by
9.9% in terms of an average stress under the channel. A 0.2 \mu m stressed nitride layer, on the other hand, increases the STI width impact by 12.9%. A 2 GPa liner can increase impact by 10.3%. These changes seem negligible and also indicate that STI stress width effect will still be important across such process variants. The silicon data reported in [30] supports our findings, i.e., STI stress width effect is still important even in the presence of stressed nitride liners. The nominal stress values usually increase for the $S_{yy}$, which is shown in Figure 3, as this component is in the direction of the nitride liner. The $S_{xx}$ stress component typically reduces with these changes.

![Fig. 5. Stressed nitride liner as mobility enhancer.](image)

**TABLE V**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>SA</th>
<th>SB</th>
<th>STIWidth</th>
<th>STIHeight</th>
</tr>
</thead>
<tbody>
<tr>
<td>STIHeight</td>
<td>0.2, 0.4 \mu m</td>
<td>0.2, 0.4 \mu m</td>
<td>0.1, 0.2 \mu m</td>
<td>0.1, 0.2 \mu m</td>
</tr>
</tbody>
</table>

**Impact of STI Height.** Parameters used to analyze the impact of STI height on stress are shown in Table V. The height of the STI trench is changed to see the impact of STI height on stress. We have observed that increasing the STI height can result in a reversal of the STI stress, i.e., changing it from compressive to tensile for the $S_{yy}$ component. This observation is in line with what has been presented in [31]. Comparison of average stress values shows a reduction of up to 6% for the STI width impact due to variation of STI height. As with stress liner impact, we believe that our conclusions below remain valid across STI heights used by different processes.

Analyses and optimizations proposed in the rest of the paper are not tethered to the models developed in this section, and can be used with other models after appropriate modifications. For example, there are known STI processes which may induce tensile instead of compressive stress. This may be due to STI trench height, or material and thermal processing differences, such as HDP (High Density Plasma) CVD (Chemical Vapor Deposition) as used in [9]. The optimization procedure presented below can be adopted to such a scenario with minor changes. Furthermore, the proposed models show monotonic response with respect to the STI proximity and widths. This results in an optimization scheme where maximum or minimum allowed dimensions will improve performance. With models that are non-monotonic, the optimization algorithm would need to be altered to provide an optimal solution.

Even as new mobility enhancement techniques show substantial variation in mobility, STI has been a mainstream methodology for over a decade. STI processes are much better controlled than the new stress engineering methods. Hence, there is much smaller observed variability in the STI mobility impact. Lithography effects will show negligible impact on the mobility change due to STI, as long as the layout is designed considering DFM rules such as using regular active regions without unnecessary corners.

Finally, since active-layer fill insertion is a knob that we propose to exploit, we comment on additional process considerations for active-layer filling.

- There exist design rules which restrict the maximum active layer density, with this constraint arising for reasons of STI CMP uniformity. Such rules must be observed.
- Insertion of active-layer fills can potentially increase the total capacitance of inter-cell M1 routing, and may induce additional RCX modeling and characterization for a given process technology. However, our methodology should not affect extraction of intra-cell M1 routing; as our active-layer fills are floating, their impact is smaller.
- Reduced STI widths may slightly increase leakage between NMOS and PMOS transistors as well as between devices of the same type. However, the active to active design rules are typically set such that this leakage is minimized to a negligible level.

**III. STRESS-AWARE TIMING ANALYSIS**

In this section we describe our STI stress-aware timing analysis methodology. We adapt the traditional SPICE-based timing analysis flow to consider stress induced by STI widths.

**A. Traditional SPICE-Based Timing Analysis**

Cell-level static timing analysis tools such as Prime-Time offer a good tradeoff between accuracy and analysis speed. Full designs or their blocks are typically analyzed and signed off with circuit-level static timing analysis (STA). However, if greater accuracy is desired, SPICE-based analysis, which has better accuracy but substantially slower analysis speed, is employed. Since running full-chip SPICE analysis is not feasible, critical paths are first identified with static timing analysis and then simulated with SPICE.

A typical netlist input to SPICE is layered into the following three tiers:
- **Device-level models** which contain transistor parameters in the form of coefficients of functions defined in BSIM or equivalent formats. Device-level models allow output waveforms for PMOS and NMOS devices to be simulated.

- **Cell-level netlists** which describe the connectivity of the devices that comprise individual cells. Cell-level netlists instantiate device-level models and allow SPICE to simulate waveforms at the outputs of cells in the library when subjected to a stimuli.

- **Critical path netlists** which describe the connectivity between the cells for each critical path. Critical path netlists instantiate cell-level netlists and can be simulated to calculate the delay of the critical paths.

As noted above stress-induced device mobility change is determined by: (1) the separation between the gate and the active edges, and (2) by the size of the STI region that surrounds the active region of the device. Fortunately, the separation between gate and active edges is fixed when the cells are designed, and the contribution of this separation to stress and mobility can be modeled at the cell level. Specifically, in the BSIM 4.3.0 device-level models, stress parameters \( S_A \), \( S_B \), and \( S_C \) have been introduced to model the stress effect as a function of gate and active edge separation. In cell-level netlists these parameters are passed with the instantiation of the device-level models. Cell-level netlists are used in library characterization to generate gate-level timing models for use in STA. An example of device-level instantiation with stress parameters is shown in Figure 6.

```spice
.model NCH NMOS
.MM2 D G S B
.MM1 D G S B
NCH SA=0.2u SB=0.2
PCH SA=0.19u SB=0.19u
.ends

.subckt INVX1 A Z
  .subckt INVX1 A Z
    .param NMOB = Our_NMOS_Model (PL, PR, NL, NR)
      MM1 D G S B
        MM1 D G S B
          NCH SA=0.2u SB=0.2
          PCH SA=0.19u SB=0.19u
          .ends
          .param PMOB = Our_PMOS_Model (PL, PR, NL, NR)
            MM2 D G S B
              MM2 D G S B
                NCH SA=0.2u SB=0.2
                PCH SA=0.19u SB=0.19u
                MOB=NMOB
                MOB=PMOB
          .ends
    .param PL=0.08u PR=4.08u NL=0.06u NR=4.06u
    .param PL=2.1u PR=5.0u NL=2.04u NR=5.0u
  .ends

.subckt NAND2X1 N2 1 N2 NAND2X1
  .param PL=0.08u PR=4.08u NL=0.06u NR=4.06u
  .param PL=2.1u PR=5.0u NL=2.04u NR=5.0u
  .param PL=5.0u PR=5.0u NL=5.0u NR=5.0u
  .ends

.subckt BUFFX1 N3 N4 BUFFX1
  .param PL=0.08u PR=4.08u NL=0.06u NR=4.06u
  .param PL=2.1u PR=5.0u NL=2.04u NR=5.0u
  .param PL=5.0u PR=5.0u NL=5.0u NR=5.0u
  .ends

.model NCH NMOS
  *Other stress parameters defined

.ends
```

**Fig. 6.** Instantiation of device-level models in a standard cell SPICE netlist. The parameters added in BSIM 4.3.0 to partially model stress are shown in bold.

The stress effect due to STI width is not modeled primarily for the following two reasons:

- STI width is determined by the placement of the cells, so that stress effect due to STI cannot be captured in library characterization. A new methodology that analyzes a placed design and annotates STI width information for use in timing analysis is required.

- Stress effect due to STI is of smaller magnitude than gate and active edge separation.

**B. STI Stress-Aware Timing Analysis**

Our approach analyzes the placement of a design and the standard cell layouts to calculate the STI widths for all critical cells in the design. The STI widths are then passed as parameters which are used in the models developed in the previous section.

We modify the cell-level netlists such that parameters \( PL, PR, NL, NR \) which capture the STI width are passed to them. Parameter \( PL \) is the spacing between the boundary of a cell and the neighboring active region to the left of its positive active region. Similarly parameter \( PR \) is the spacing between the boundary of a cell and the neighboring active region to the right of its positive active region (PRX). Parameters \( NL \), and \( NR \) are similarly defined for negative active regions (NRX). The parameters are set in the critical path netlists when cells are instantiated as shown in the example in Figure 7.

```spice
.subckt INVX1 A Z
  .param NMOB = Our_NMOS_Model (PL, PR, NL, NR)
    MM1 D G S B
      MM1 D G S B
        NCH SA=0.2u SB=0.2
        PCH SA=0.19u SB=0.19u
        MOB=NMOB
        MOB=PMOB
    .ends

.subckt NAND2X1 N2 1 N2 NAND2X1
  .param PL=0.08u PR=4.08u NL=0.06u NR=4.06u
  .param PL=2.1u PR=5.0u NL=2.04u NR=5.0u
  .param PL=5.0u PR=5.0u NL=5.0u NR=5.0u
  .ends

.subckt BUFFX1 N3 N4 BUFFX1
  .param PL=0.08u PR=4.08u NL=0.06u NR=4.06u
  .param PL=2.1u PR=5.0u NL=2.04u NR=5.0u
  .param PL=5.0u PR=5.0u NL=5.0u NR=5.0u
  .ends

.model NCH NMOS
  *Other stress parameters defined

.ends
```

**Fig. 7.** Critical paths instantiate cell-level netlists which instantiate device-level models. Our modifications to the traditional flow to model STI width dependent stress are shown in bold.

The \( PL, PR, NL, NR \) parameters can be calculated from the placement and the cell’s layouts, specifically, the cell boundary to active spacings. Computation of \( PL \) for a cell, which is the spacing between the cell’s boundary and the positive active region of the cell to its left, is as follows. The spacing between the cell and its left neighboring cell is found from the placement. The spacing between the positive active region of the neighbor and its cell boundary is found from layout analysis of the neighbor. The two spacings are then added, with correct consideration of the orientations of the cell and its neighbor. Other parameters \( PR, NL, \) and \( NR \) are calculated similarly. Figure 8 illustrates the calculation.
We note that our flow needs modifications to work for cells with complex active shapes such as flip-flops and multiplexors. Active shape complexities include non-rectangular shapes and non-continuous shapes. To model STI stress impact for non-rectangular active shapes, modifications such as those employed by BSIM to handle non-rectangular active may be used. For cells with non-continuous active shapes, devices can be completely shielded from STI width outside the cell and our flow should not alter their mobility. In our analysis and optimization, we focus on the cells with simple active shapes and do not change the mobilities for cells with complex active shapes (i.e., use traditional analysis and no optimization for them). Fortunately, the most frequently cells such as inverters, buffers, NAND’s, NOR’s, AND’s, and OR’s have simple active shapes so we consider and optimize most cells in our designs.

**C. Alternative Flow**

STI stress aware timing analysis can also be performed by cell-level STA. Towards this standard cells in the library can be characterized for different STI width configurations around them. Since stress dependence on STI width is relatively gradual, STI widths can be binned into a small number of bins to reduce the total number of STI width configurations. For each standard cell, variants may be created corresponding to each STI width configuration. Given the STI width, models presented in the previous section are used in library characterization. The STI width of a cell in a design can be computed from the placement and standard cells layouts, and can be used to find the variant that has the closest STI width configuration. The cell can then be bound to the variant in the library and cell-level STA run to perform STI stress aware timing analysis.

**IV. TIMING OPTIMIZATION**

In this section we present our timing optimization methodology. The basic idea exploited in our optimization is that STI widths of devices can be altered to change their mobility and improve performance. Specifically, the alteration involves increasing the STI widths for PMOS devices and decreasing them for NMOS devices. We identify the timing critical cells and alter their STI widths to improve the circuit performance. In our approach we use the following two knobs to alter the STI widths:

- Placement perturbation. The placement of a layout can be changed to increase or decrease the spacing between neighboring cells which directly increases or decreases the STI width. Additionally, spacing cells apart can allow fills, for which initially there was insufficient space, to be inserted.
- Active-layer fill (RX fill) insertion. Active-layer fills are rectangular dummy geometries inserted on the active (RX) layer primarily to improve planarity after chemical mechanical polishing (CMP). However, such geometries also reduce the STI width of the devices next to which they are inserted. The STI width after insertion of an RX fill next to a device is the spacing between the active region of the device and the fill.

We now present the details of the above two knobs.

**A. Active Layer (RX) Fill Insertion**

Even though RX fills are non-functional geometries, their effect on the stress is identical to that of active regions of devices. When inserted next to the active region of an NMOS device, fills substantially reduce the STI width and stress of the device, and consequently improve the performance of the NMOS device. On the other hand, fills inserted next to a PMOS device reduce STI width and stress but consequently degrade the performance. Hence, inserting fill next to the NMOS devices but not next to the PMOS devices of a cell improves performance.

Circuit delay improves when the delay of setup-critical cells is reduced. So, we insert rectangular RX fills next to the NMOS devices, to the left and right of the cell. No RX fills are inserted next to the PMOS devices; so the PMOS remains exposed to a large STI width and stress. The devices closer to the active boundary experience the maximum benefit of this optimization. Since the most frequently used cells in the designs are small, a large fraction of devices in the design benefit from fill insertion. Our technique can also be employed for hold-time critical cells in the reverse manner, i.e., insert fills next to the PMOS devices but not next to NMOS devices to slow down the cell.

Figure 9 shows an example standard cell with PRX (active regions for PMOS devices) and NRX (active regions for NMOS devices). As can be seen, active regions exist under the top and bottom cell boundary that completely shield the cell from STI stress effects in the direction orthogonal to the carrier (current) flow.
direction. Hence, we only apply our optimization in the parallel direction by inserting fill to the right and left of a cell. Figure 10 illustrates fill insertion for a setup-critical cell; NRX fills are inserted next to the NRX region to reduce stress and fasten the NMOS devices. Figure 11 illustrates the approach for a few setup-critical cells in a standard cell row.

![Cell Boundary PRX NRX](image)

**Fig. 9.** A generic standard cell with polysilicon, positive active regions (PRX), negative active regions (NRX), and cell boundary shown.

![NRX FILL](image)

**Fig. 10.** The generic cell of Figure 9 optimized with fill insertion for setup-time criticality.

![A B C D E](image)

**Fig. 11.** A row of standard cells after active-layer fill insertion for setup-time improvement. Cells patterned with diagonal lines are the setup-critical cells and filled rectangles are the inserted active layer fills.

All fills are inserted subject to the design rule constraints (DRCs) and introduce no DRC violations. We have already noted that no additional mask step is required, and that M1 capacitance impact is likely negligible. Since the fill insertion knob can only decrease STI width, NMOS performance can be improved but PMOS performance can at best be kept constant. However, neighboring cells which have very small spacing and between which fills cannot be inserted can be spaced apart by placement perturbation to allow fills to be inserted.

**B. Intra-Row Placement Optimization**

We now present the placement perturbation knob which can increase (decrease) the STI width and improve PMOS (NMOS) performance. Placement of a cell determines its location (consequently its neighbors and spacings with them) and its orientation. In our optimization we change the locations of the cells such that spacings are altered but the ordering of cells in a standard cell row is not affected. Increased spacing next to a cell, increases the STI width and improves the delay of the PMOS devices. However, the delay of the NMOS devices increases with increased spacing. Fortunately, we can utilize our first knob, RX fill insertion, to reduce the NMOS STI width and improve its delay as well. In fact, if the spacing between cells is too small for fill insertion, placement can facilitate fill to be inserted by creating additional space for it. The placement perturbation just reorganizes the whitespace in the standard cell row of the cell without requiring any additional space.

**Minimizing delay increase due to wirelength increase.** The perturbation of detailed placement from the original placement results in small wirelength change, which can impact wire parasitics and consequently timing. Even though our localized placement perturbations do not significantly affect timing, small changes in the timing of critical paths can affect the minimum clock cycle time. To minimize the timing change of critical paths, we fix the cells and nets in the critical paths. Fixed cells are not moved during optimization and fixed nets are not changed during engineering change order (ECO) routing that is performed after optimization. Since the nets in the critical path are fixed, all cells connected to these nets should also be marked as fixed and not moved during optimization. We note that the delay of such nets can marginally change due to the coupling capacitance with neighboring nets, the routing for which may change. We also fix all flip-flops, clock buffers, and clock nets to avoid any impact on the clock tree. So our list of fixed cells comprises of timing critical cells, their fanout cells, flip-flops, and clock buffers.

Our intra-row placement optimization attempts to create space on the right and left sides of each timing critical cell. In the process, the minimum number of cells are displaced to minimize the wirelength impact. Figure 12 presents the pseudo-code for our intra-row placement optimization. For each timing critical cell, right and left spacings are increased by functions `createRightSpace` and `createLeftSpace` respectively to attain a spacing of up to $S$. The spacing, $S$, may not always be attainable because of the presence of fixed cells and availability of limited space in the row. For the right side, the function `cellsToMoveRight` finds the minimum number of cells to move. Then the function `moveCellsRight` flushes the computed number of cells to the right as much as possible.

Our algorithm sequentially processes critical cells in
Input: Placed design; set of timing critical cells, \( T \); set of fixed cells, \( F \); maximum spacing to create, \( S \)

Output: New placement with altered inter-cell spacings

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>forall cells ( t \in T )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>createRightSpace( (t) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.2</td>
<td>createLeftSpace( (t) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>createRightSpace( (t) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>( n = \text{cellsToMoveRight}(t) );</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>\text{moveCellsRight}(t, n);</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cellsToMoveRight( (t) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>( i \leftarrow t );</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>( j \leftarrow \text{cellToRightOf}(t); )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>( \text{accumulatedSpacing} = 0; )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>\text{cellsToMove} = 0;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>while ( \text{accumulatedSpacing} \leq S ) and ( j \notin F )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.1</td>
<td>( \text{accumulatedSpacing} += \text{interCellSpacing}(i, j); )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.2</td>
<td>\text{cellsToMove}++;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.3</td>
<td>( i \leftarrow j )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.4</td>
<td>( j \leftarrow \text{cellToRightOf}(i); )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>return \text{cellsToMove};</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>moveCellsRight( (t, n) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>// flush ( n ) cells to the right of Cell ( t ) towards the right to create space ( S )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>( F \leftarrow F \cup { \text{n cells to the right of Cell } t } )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>createLeftSpace( (t) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>// similar to createRightSpace( (t) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 12. Pseudo-code for intra-row placement optimization.

descreasing order of their criticality. Cells displaced in an iteration to create space are added to the list of fixed cells to lock them for successive iterations. This can limit the optimization of critical cells processed later in the algorithm. Therefore, we run the algorithm multiple times with increasing value of \( S \). This enhancement allows a fair distribution of whitespace among all critical cells. We increase the value of \( S \) from 0.6\( \mu \text{m} \) to 1.8\( \mu \text{m} \) in steps of 0.2\( \mu \text{m} \). Starting with a smaller value of \( S \) leads to a more equitable distribution of whitespace at the expense of runtime. For designs with high utilization ratios, starting \( S \) of less than 0.6\( \mu \text{m} \) may be desirable. We have found that the STI width effect saturates at 1.8\( \mu \text{m} \) and there is negligible change in stress beyond that.

Our second enhancement is perturbing the critical cells to balance the space on the right and left sides of them. Since the stress effect decays rapidly with space, nearly-equal spacings on both sides are desirable. We limit the perturbation to 0.6\( \mu \text{m} \) to minimize wirelength and the associated delay increase. The space required to insert RX fill is typically very small and in the 0.2\( \mu \text{m} \) range. Therefore, if the optimization creates any space for PMOS optimization, fill can always be inserted to improve the deteriorated NMOS performance. Figure 13 illustrates placement perturbation and fill insertion for setup-time optimization on a standard cell row.

Fig. 13. Placement change and fill insertion for setup-time optimization. A standard cell row is shown before optimization, after placement perturbation, and after fill insertion. Cells patterned by diagonal lines are the setup-critical cells for which timing is optimized. Fixed cells are patterned with the brick pattern and their placement cannot be changed.

While it is possible to perform fill insertion without placement perturbation, we have found the associated performance benefits to be very small. Both knobs complement each other: placement creates space for fill insertion and fill insertion improves the performance of the NMOS devices that are slowed down by placement perturbation.

Our overall STI stress-aware placement and fill optimization flow is as follows:
1) Identify critical paths and critical cells
2) Perform intra-row placement optimization
3) Perform ECO routing followed by parasitic extraction
4) Perform RX fill insertion
5) Evaluate the optimized layout with STI stress-aware timing analysis

V. EXPERIMENTAL STUDY

We now present our experiments to evaluate the proposed optimization methodology. Our experiments assess the impact of our optimization on the minimum clock cycle time, delay of top critical paths, and final routed wirelength.

A. Experimental Setup

The details of the testcases used in our experiments are presented in Table VI. We use Synopsys Design...
There is no image attached to the given text, but it appears to be a continuation of the previous discussion on cell optimization and placement perturbation. The text mentions the use of stress-aware timing analysis to accurately report the minimum cycle time (MCT) and total path delay (TPD) for a circuit, and it also refers to previous work on stress optimization and its impact on circuit performance.

The text describes how stress-aware timing analysis reduces pessimism in analysis by explicitly accounting for STI stress during optimization. It notes that stress-aware analysis reduces MCT by 5.75%, and TPD by 5.28% on average. We use stress-aware analysis to evaluate our optimization in the remainder of this section.

In Section IV, two optimization knobs are presented: fill insertion and placement perturbation. Although these techniques complement each other, they are evaluated separately. Placement perturbation, without fill insertion, is not interesting because it slows down the NMOS devices while speeding the PMOS. Table VIII presents the improvements in MCT and TPD due to fill insertion. Since we optimize several critical paths, TPD reduces. However, we observe that reductions in MCT and TPD are typically under 1%.

We now evaluate the simultaneous use of the proposed placement perturbation and fill insertion knobs. In addition to comparing MCT and TPD results, we also compare the wirelength which changes because of placement perturbation. After placement perturbation, several nets are left dangling; we perform ECO routing to route them, and follow by RC extraction and stress-aware timing analysis to accurately report the MCT and TPD results for the optimized case. The runtime of our placement and fill optimization is generally small; it depends on the circuit size and the number of critical paths to be optimized. In our experiments, the runtime was under one minute for all testcases on a 2.2GHz AMD Opteron/8GB RAM machine running Linux 2.6.

Table IX presents our results for our four testcases. For negligible increase in wirelength, we observe 4.37% and 5.15% reductions in (stress-aware) MCT and TPD averaged over the testcases C5315, ALU, and AES. The testcase S38417, however, demonstrates smaller improvements. We attribute this to the fact that S38417 is an artificial testcase with over 50% of its cells being flip-flops. We do not allow our optimization to change the locations of flip-flops, so as to avoid changes to the clock tree; hence, in the S38417 testcase, we can perturb the placement of fewer cells.

Figure 14 shows the histograms for the delays of top 200 critical paths of our testcases AES before and after optimization. As can be seen, the delay distribution has shifted to the left (lower delay) substantially.

We also tried the technique to optimize hold-critical paths but found negligible improvement in hold slack for our testcases. This is because stress optimization can only change cell delays by 10%-20% and for hold-critical paths the cell delays are very small. The change in the delay of hold-critical paths is insignificant with our approach and traditional delay introduction methods such as insertion of delay elements or wire snaking must be used.
VI. CONCLUSIONS

We have conducted TCAD process simulations to generate models that relate the dependence of transistor mobilities to stress induced by STI width. We have proposed an STI width-aware design methodology for standard cell place-and-route designs. The proposed stress-aware timing analysis technique reduces pessimism in delay analysis. Over traditional corner-based analysis, delays reported by stress-aware analysis were on average 5.75\% lower. We have also devised an optimization methodology, based on cell placement perturbation, to create extra space around critical cells; this is followed by dummy diffusion insertion. The proposed optimization flow, while demonstrated with our models, can be generalized to other STI stress models. We have applied the proposed optimization flow on a number of testcases implemented with industry 65nm libraries. Our data shows that STI width optimization can increase performance by 2.44\% to 5.26\% with no area penalty. The proposed optimization can form the basis of circuit optimization that exploits upcoming stress-engineered transistor technologies in 65nm and below processes.

VII. ACKNOWLEDGMENTS

We would like to thank Prof. Hoong-Joo Lee of Sangmyung University, Korea, and Frank Geelhaar of AMD for useful discussions, and Swamy Muddu for help with the implementation and the testcases.

REFERENCES

Symposium on VLSI Technology, 2005.


2004.


Oxide for Quarter Micron Shallow Trench Isolation (STI),” 
Symp. on VLSI Technology Digest of Technical Papers, pp. 
158-159, 1996.

Dopant Diffusion in Scaled MOSFETs,” IEEE Tran. on Elec- 

Induced by STI in Short MOSFETs Using Externally Applied 
2004.

“Piezoresistive Characteristics of Short-Channel MOSFETs on 

Effect on Layout Dependence of MOS Electrical Characteristics,” 

Transistor Performance,” Int. Conf. on Simulation of Semicon- 
ductor Processes and Devices, 2005.


“Stress-Aware Design Methodology,” Int. Symposium on Qual- 

[28] Y.-M. Sheu et al., “Modeling Well Edge Proximity Effect on 
Highly-Scaled MOSFETs”, IEEE Custom Integrated Circuits 
Conference, 2005.

[29] M. Miyamoto, H. Ohta, Y. Kumagai, Y. Sonobe, K. Ishibashi 
and Y. Tainaka, “Impact of Reducing STI-Induced Stress on 

FET Characteristic Fluctuation Caused by Layout Variation,” 

[31] R. Arghavani et al., “Stress Management in Sub-90nm Tran- 
sistor Architecture,” IEEE Tran. on Electron Devices, vol. 51, 