Incremental Detailed Placement for VLSI Design
Prabhakar R¹, Sreenivasa Murthy K E ², Soundara Rajan K ³
Associate Professor¹, ECE Dept, Dr KVSRT, Kurnool, A P, India.
Principal², SSITS, Rayachoti, Kadapa, A P, India.
Former Rector & Reid. Professor³, JNTUA, Ananthapuramu, A P, India.

Abstract: The main purpose of VLSI placement is to place the objects into fixed chip such that there should be no overlaps among the objects and some cost metric such as wire length and routability is optimized. Physical synthesis optimizations and changing the placement method typically change the locations of cells, resize cells or add more cells to the design after global placement. But, those changes generally leads to wire length increases; thus another method of optimizations to further improve wire length, timing and routing congestion characteristics is required. The Incremental Detailed Placement techniques could be useful in this condition. So, we propose a new detailed placement paradigm, which use a set of pin-based timing and electrical constraints in detailed placement to prevent it from degrading timing or violating electrical constraints while reducing wire-length.

Key words: Routability, Physical Synthesis, Worst Negative Slack, figure-of-Merit, Routing Congestion.

1. INTRODUCTION
Global placement is one of the most typical processes in modern physical design. Its task is to determine the overall locations of cells in the design. However, physical synthesis designs such as buffering and gate sizing are applied after component placement to further optimize timing. These methods usually insert new cells or change the size of existing cells. Engineering change order (ECO) is another source of modifications for designs under optimization. It could also introduce new logics, change the physical sizes of objects, or change the locations of existing cells. All these changes may probably result in overlaps among cells. Therefore, this design needs additional (extra) legalization to remove those overlaps. Even though many legalization methods have been proposed to minimize the disturbance to the original placement, they usually result in wire length degradation.

The Detailed placement techniques such as Simulated Annealing (SA) based swapping and moving [7], cell interleaving [12], branch-and-bound reordering, branch-and-price reordering, guided local search, global swap and local reordering, and net length constrained SA approach can all reduce the total wire length only. But, reducing total wire length do not necessarily result in timing improvement, particularly after physical design. Therefore an efficient delay model is required for Incremental detailed placement to avoid doing any harm to timing critical paths while improving total wire length.

1.1 PIN BASED TIMING AND ELECTRICAL CONSTRAINTS
In this section, we will model timing and electrical constraints for Incremental Detailed Placement. The purpose of adding these constraints is to prevent any degradation of timing results (worst negative slack, total negative slacks, etc). Which uses constraints on timing paths and our timing constraints are imposed on individual pins. Even though these constraints may be preserved using pin-based constraint, which would greatly simplify the timing computation during placement because the expensive path propagation computation is not required.

1.2 Delta Arrival Time Constraints
The arrival time of each pin of a gate is defined as the addition of delay segments from timing start points, i.e. PIs or the output of a sequential logic, to the pin itself. The arrival time of each gate is simply defined as the addition of delay segments on the most critical input pin.
Thus, let Nm be a set of gates or PIs connected to the input of gate m, and gate k ∈ Nm connected to input pin j of gate m. The arrival time of pin will be

$$\Delta T_{m,j} = \Delta T_{k} + d_{k,m}$$

(1)
The delta arrival time of input pin as the differences between the arrival time of pin itself and the arrival time of the gate. As shown in Fig 1, the delta arrival time of the input j of gate m, \(\Delta \Delta T_{m,j}\) is defined as follows:

$$\Delta \Delta T_{m,j} = \Delta T_{m} - \Delta T_{m,j}$$

(2)
The delta arrival time for a primary output pin (PO) or a sequential logic input pin is always zero because there is only one pin to compare with. But the delta arrival time is absolutely different from slack. Slack is the difference between arrival time and required arrival time, while delta arrival time is the difference
between the arrival time of an input pin to the most critical input pin. Pins with same slack can have different delta arrival time, and pins with same delta arrival time can have different slack.

\[ \Delta T_{m} = \Delta T_{j} + d_{m,j} + \Delta \Delta T_{m,j} \]  

(3)

If the placement of the cells changes, the gate delay \( d_{m} \), wire delay \( d_{m,j} \) and arrival time on all the gates will also change. Thus, the new arrival time on gate \( m \) can be computed as a new formula, which is shown in below:

\[
\text{new}_{\Delta T_{m}} = \max_{k \in N_{m}} \left[ \text{new}_{\Delta T_{k}} \right] + d_{k,m} + \Delta d_{k} + \Delta \Delta d_{k,m} \]  

(4)

![Fig 1: example of Delay Model](image)

Fig 1 shows a small network with 4 PI, 1 PO, 3 gates and 7 nets.

The upper figure 1(a) shown the original gate and wire delays, arrival times on all three gates, PIs and PO, and delta arrival times on noncritical internal pins, while the lower figure 1(b) shown the changed gate/wire delays, and new arrival times. We have observed that the increase on the merged gate delay and wire delay is less than the delta arrival time. For example, the combined gate and wire delay between gate B and C changes from 2+1=3 to 4+6=10. However, the \( \Delta \Delta T \) on this connection is 9, which is still greater than the amount of delay increase 10-3=7. Even the critical input pin of gate C changes, the arrival time on the input of C is reduced from 18 to 16. We have also verified that the arrival times on all the gates do not increased and so does the PO arrival time. Therefore the slack on PO is not degraded.

2. INCREMENTAL DETAILED PLACEMENT

In this design we used the Incremental Detailed Placement method that convert placement from one legal solution to another legal solution. These methods take a legally placed net list, change locations of cells while still maintaining the legality. Normally these approaches only check whether the movements reduce the total wire length or not.

2.1 Constraint Formulation

During Incremental Detailed Placement, we can obtain an accurate estimation of the total half perimeter wire length (HPWL). If we can roughly estimate delay or slew based on HPWL, then we can verify whether the constraints are satisfied or not.

To do this we used a differential gate delay and wire delay model, which estimates delay and slew increments of placement change. Conversely, the main difference is the gate delay modeling. Therefore the gate delay and output slew is only determined by output load. The advantage of this is to avoid slew propagation, which is time consuming because which has to propagate the slew all the way down to the end and re-compute the timing on many cells down the way. Using a conservative gate delay modeling actually makes the timing constraints more conservative, which helps protecting of any timing degradation.

Gate delay and output slew can be represented as linear functions of input slew and output load as follows:

\[
d_{k} = A_{0} + A_{1}c_{k} + A_{2}s_{k,j} \]  

(5)

\[
s_{k} = B_{0} + B_{1}c_{k} + B_{2}s'_{k,j} \]  

(6)

Where \( d_{k} \) and \( s_{k} \) are the delay and output slew of gate \( k \); \( s_{k,j} \) is the most critical input pin of gate \( k \) and \( s'_{k,j} \) is the input slew on pin \( j \). \( A0, A1, A2, B0, B1, B2 \) are constants determined by the standard cell library characterization. Since we have assumed the critical input pin slew is constant, the differential gate delay and output slew can be computed by as follows:

\[
\Delta d_{k} = A_{1}\Delta c_{k} \]  

(7)

\[
\Delta s_{k} = B_{1}\Delta c_{k} \]  

(8)

Where \( \Delta d_{k} \) and \( \Delta s_{k} \) is the gate delay and output slew increments for gate \( k \), respectively. \( \Delta c_{k} \) is the total output load increment, which can be computed by

\[
\Delta c_{k} = c\Delta l_{i} \]  

(9)

Where \( c \) is the unit wire capacitance; \( \Delta l_{i} \) is the total Wire length (HPWL) increment for net \( i \), which gate \( k \) drives.

2.2 Objective Function

The objective function of our Incremental Detailed Placement is to reduce both TWL and TNS. Assuming and we can guarantee that slacks do not degraded. We can use weighted total wire length as the optimization objective for Incremental Detailed Placement. Critical nets (nets with negative slacks) are given higher weights than other nets. The weighted wire length objective function is given below.

\[
W_{TWL} = \sum w_{i}l_{i} \]  

(10)

Where \( W_{TWL} \) is weighted total wire length. \( w_{i} \) is the net weight for net \( i \), and \( l_{i} \) is the HPWL of net \( i \).
3. EXPERIMENTAL RESULTS

We have implemented the Incremental Detailed Placement in DSCH and MicroWind Tool. The technology for each design is reported in Table I. The worst negative slack (WNS) and Figure-of-Merit (FOM) of these designs are reported in Table II.

Table I: Design, Technology, TWL and Power of IDP:

<table>
<thead>
<tr>
<th>Design</th>
<th>Technology</th>
<th>TWL (µm)</th>
<th>Power (µW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D Flip Flop</td>
<td>90-nm</td>
<td>791.58</td>
<td>0.326</td>
</tr>
<tr>
<td>Right Shift Register</td>
<td>90-nm</td>
<td>15796.15</td>
<td>0.938</td>
</tr>
<tr>
<td>Dual Port RAM</td>
<td>90-nm</td>
<td>3986.01</td>
<td>1.018</td>
</tr>
<tr>
<td>Synchronous Counter</td>
<td>90-nm</td>
<td>71203.59</td>
<td>25.83</td>
</tr>
</tbody>
</table>

Table II: Design, Cells, Nets, Slack & FOM of IDP:

<table>
<thead>
<tr>
<th>Design</th>
<th>No.of Cells</th>
<th>No.of Nets</th>
<th>Slack (WNS) (ps)</th>
<th>FOM (Ps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D Flip Flop</td>
<td>05</td>
<td>17</td>
<td>1.650</td>
<td>28.05</td>
</tr>
<tr>
<td>Right Shift Register</td>
<td>56</td>
<td>80</td>
<td>3.083</td>
<td>246.66</td>
</tr>
<tr>
<td>Dual Port RAM</td>
<td>12</td>
<td>44</td>
<td>2.740</td>
<td>120.49</td>
</tr>
<tr>
<td>Synchronous Counter</td>
<td>178</td>
<td>129</td>
<td>0.842</td>
<td>108.70</td>
</tr>
</tbody>
</table>

Table III shows the total Wire length (TWL) comparison of Existing Placement (Existing) method and those after TDIP, NAIP, DBIP and IDP. Considering that the placement is already optimally placed by a global placer during physical synthesis, the improvements are significant. We have highlighted those cases where TWL did not increase from existing. We have shown that the IDP improves the TWL on all placement methods with an average of 3.12% improvement, while TDIP, NAIP and DBIP improves the TWL a less.

Table III: Comparison of Total Wire length of TDIP, NAIP, DBIP & IDP with Existing Methods:

<table>
<thead>
<tr>
<th>Design</th>
<th>Existing</th>
<th>TDIP</th>
<th>NAIP</th>
<th>DBIP</th>
<th>IDP</th>
<th>Change in WL</th>
</tr>
</thead>
<tbody>
<tr>
<td>D Flip Flop</td>
<td>844.23</td>
<td>834.6</td>
<td>825.6</td>
<td>808.92</td>
<td>791.58</td>
<td>1.14%</td>
</tr>
<tr>
<td>Right Shift Register</td>
<td>16071.03</td>
<td>16016.05</td>
<td>15924.43</td>
<td>15901.76</td>
<td>15796.15</td>
<td>0.34%</td>
</tr>
<tr>
<td>Dual Port RAM</td>
<td>4094.79</td>
<td>4079.25</td>
<td>4040.4</td>
<td>4009.32</td>
<td>3986.01</td>
<td>0.38%</td>
</tr>
<tr>
<td>Synchronous Counter</td>
<td>72564.71</td>
<td>71799.08</td>
<td>71628.94</td>
<td>71374.92</td>
<td>71203.59</td>
<td>1.1%</td>
</tr>
</tbody>
</table>

Average reduction in WL

0.74% 1.41% 2.24% 3.12%

Table IV gives the Worst Negative Slack (WNS) comparison among TDIP, NAIP, DBIP and IDP. In terms of Worst Negative Slack the IDP has got better Improvement than among all methods, which is higher 61.15% than TDIP, 36.04% than NAIP, 12.72% higher than DBIP.
Table IV: Slack (Ps) of TDIP, NAIP, DBIP & IDP with Existing Method:

<table>
<thead>
<tr>
<th>Design</th>
<th>Slack(WNS) (Ps)</th>
<th>Improvement (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Existing</td>
<td>TDIP</td>
</tr>
<tr>
<td>D Flip Flop</td>
<td>6.624</td>
<td>4.22</td>
</tr>
<tr>
<td>Right Shift</td>
<td>82.55</td>
<td>63.54</td>
</tr>
<tr>
<td>Register</td>
<td>33.18</td>
<td>32.23</td>
</tr>
<tr>
<td>Dual Port RAM</td>
<td>58.98</td>
<td>26.63</td>
</tr>
<tr>
<td>Synchronous</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Counter</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig 3: Percentage of Improvement in TWL of TDIP, NAIP, DBIP & IDP with Existing Method:

TDIP, 36.02 % better than NAIP & 12.7 % better than DBIP.

Table V gives the results of Figure-of-Merit (FOM) of TDIP, NAIP, DBIP & IDP with Existing Methods. The IDP provides better FOM than all placement Methods. Which is 61.16 % better improvement than
Table V: FOM (Ps) of TDIP, NAIP, DBIP & IDP with Existing Method:

<table>
<thead>
<tr>
<th>Design</th>
<th>Existing</th>
<th>TDIP</th>
<th>NAIP</th>
<th>DBIP</th>
<th>IDP</th>
<th>TDIP (%)</th>
<th>NAIP (%)</th>
<th>DBIP (%)</th>
<th>IDP (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D Flip Flop</td>
<td>112.61</td>
<td>71.81</td>
<td>30.39</td>
<td>28.996</td>
<td>28.05</td>
<td>36.23%</td>
<td>73.01%</td>
<td>74.25%</td>
<td>75.01%</td>
</tr>
<tr>
<td>Right Shift Register</td>
<td>6604.2</td>
<td>5083.39</td>
<td>1914.01</td>
<td>281.44</td>
<td>246.62</td>
<td>23.03%</td>
<td>71.02%</td>
<td>95.74%</td>
<td>96.27%</td>
</tr>
<tr>
<td>Dual Port RAM</td>
<td>1459.5</td>
<td>1417.8</td>
<td>1364.90</td>
<td>474.79</td>
<td>120.49</td>
<td>2.86%</td>
<td>6.48%</td>
<td>67.47%</td>
<td>91.74%</td>
</tr>
<tr>
<td>Synchronous Counter</td>
<td>7607.3</td>
<td>3435.5</td>
<td>2509.1</td>
<td>2028.96</td>
<td>108.70</td>
<td>54.84%</td>
<td>67.02%</td>
<td>73.33%</td>
<td>98.57%</td>
</tr>
</tbody>
</table>

Average Improvement in FOM

Table VI shows the results of Power for all Placement Methods like TDIP, NAIP, DBIP & IDP with existing Methods. The IDP gives better power improvement over other methods. The IDP gives improvement than 31.52% TDIP, 21.97% than NAIP & 2.87% than DBIP

Table VI: Power (µW) of TDIP, NAIP, DBIP & IDP with Existing Method:

<table>
<thead>
<tr>
<th>Design</th>
<th>Existing</th>
<th>TDIP</th>
<th>NAIP</th>
<th>DBIP</th>
<th>IDP</th>
<th>TDIP (%)</th>
<th>NAIP (%)</th>
<th>DBIP (%)</th>
<th>IDP (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D Flip Flop</td>
<td>3.103</td>
<td>1.13</td>
<td>1.258</td>
<td>0.327</td>
<td>0.326</td>
<td>63.58%</td>
<td>59.46%</td>
<td>89.46%</td>
<td>89.49%</td>
</tr>
<tr>
<td>Right Shift Register</td>
<td>18.58</td>
<td>7.75</td>
<td>7.693</td>
<td>1.873</td>
<td>0.938</td>
<td>58.29%</td>
<td>58.59%</td>
<td>89.92%</td>
<td>94.95%</td>
</tr>
<tr>
<td>Dual Port RAM</td>
<td>25.28</td>
<td>10.12</td>
<td>5.063</td>
<td>2.537</td>
<td>1.018</td>
<td>58.97%</td>
<td>79.97%</td>
<td>89.96%</td>
<td>95.97%</td>
</tr>
<tr>
<td>Synchronous Counter</td>
<td>94.82</td>
<td>50.570</td>
<td>30.637</td>
<td>25.831</td>
<td>25.83</td>
<td>46.67%</td>
<td>67.69%</td>
<td>72.76%</td>
<td>73.18%</td>
</tr>
</tbody>
</table>

Average Improvement in Power

Fig 7: Percentage of Improvement in FOM (Ps) of TDIP, NAIP, DBIP & IDP with Existing Method:
4. CONCLUSION & FUTURE SCOPE:

Incremental Detailed placement not only reduced Total wire length, but also significantly improves timing (WNS & FOM) and Power. These constraints and objective function are simple to implement and can be applied to many detailed placement frameworks. Throughout our work we used minimum sized transistors for the Placement, further if the transistor sizes are still reduced we may get further better improvement in the results.

REFERENCES


Authors Profile:

Mr. R. Prabahakar, pursuing Ph.D in the field of VLSI in JNTUA, Andhra Pradesh, India. He is working as Associate professor in ECE Department in Dr KV Subba Reddy Institute of Technology, Dupadu, Kurnool, A P, India. He is having 17 years of experience in teaching and has been worked in various Engineering Colleges. His area of interest is VLSI, Probability Theory & Stochastic Processes, Communications and Digital IC Applications.
2. Dr. K.E.Sreenivasa Murthy, did his B.Tech (ECE) and M.Tech from Sri Venkateswara University, Tirupati and obtained his Ph.D from Sri Krishnadevaraya University, Anantapur. His interested areas are Microprocessors, Computer hardware and Digital Signal Processing. Under his guidance two M.Phil and four Ph.D are awarded. At present eight students are working under his guidance for their Ph.Ds in JNTU, SKU etc. He is having around 25 publications to his credit. At present he is working as Principal of Sri Sai Institute of Technology & Science, Rayachoti, Kadapa, A P, India.

3. Dr. K.Soundara Rajan, Former Rector & Retired Professor of JNTUA Anantapur, Andhra Pradesh India and He is Member of All India Council for Technical Education (AICTE). He worked in various positions in JNTU, India. He is having more than 35 years of experience in teaching. More than 30 Research Scholars are working under him and more than 40 students awarded their Ph.D Degrees under his guidance.