

# Load Redistribution Technique for Timing Optimization

Sunny Arora, Shilpa Gupta, Shruti Rakheja  
Freescale Semiconductors  
Microcontroller and solution group

## Abstract

*This paper describes a technique to optimise a timing critical path. This is done with the help of proposed algorithm which redistributes the load in such a manner that the cumulative delay of a segment of a path is minimum while total load remains the same. This load redistribution algorithm can be iteratively applied to the whole design for timing optimization. The delay minima is calculated through calculus. The proposed algorithm gives 11% additional saving on a highly timing optimised segment. Similar approach is applied for determination of internal power minima.*

## Introduction

With the increase in demand for higher frequency in designs, closing timing at such frequencies has always been a challenge. To address this challenge many techniques to optimise timing critical paths have been introduced. In some techniques buffers are inserted in the path to improve the transition which reduces the cumulative delay[3]. But the delay of a buffer is an overhead. There are other techniques in which we can selectively re-synthesize and place the path[1,2] but this not feasible at post layout stage.

In this paper an algorithm is proposed which uses the fact that different cells in a library

behave differently with varying load and input transition. This algorithm redistributes the load of a segment of path in such a manner that the cumulative delay is minimum while total load remains same. This distribution of the load is such that a cell which is less sensitive to increase in load is allowed to bear higher load compared to more sensitive cell. Present algorithm claims to optimise timing of a saturated path by 11% without any overhead.

The paper is organised in six sections. Sections 2, describes the basic concept with the help of a graph. Section 3 describes the proposed algorithm, 3.1 and 3.2 describes its application in terms of polynomials and three dimensional graphs. Section 4 states the results after applying the algorithm on real design and finally, section 5 concludes the paper.

## 2. Basic concept:

There are various cells in a library and delay and transition sensitivity to output load and input transition varies from cell to cell. In the below figure the graph (a) in figure 1 shows lesser sensitivity of cells to load than graph (b). The proposed algorithm redistributes the load in such a manner that the cell which is less sensitive to load has more load than the more sensitive cell.

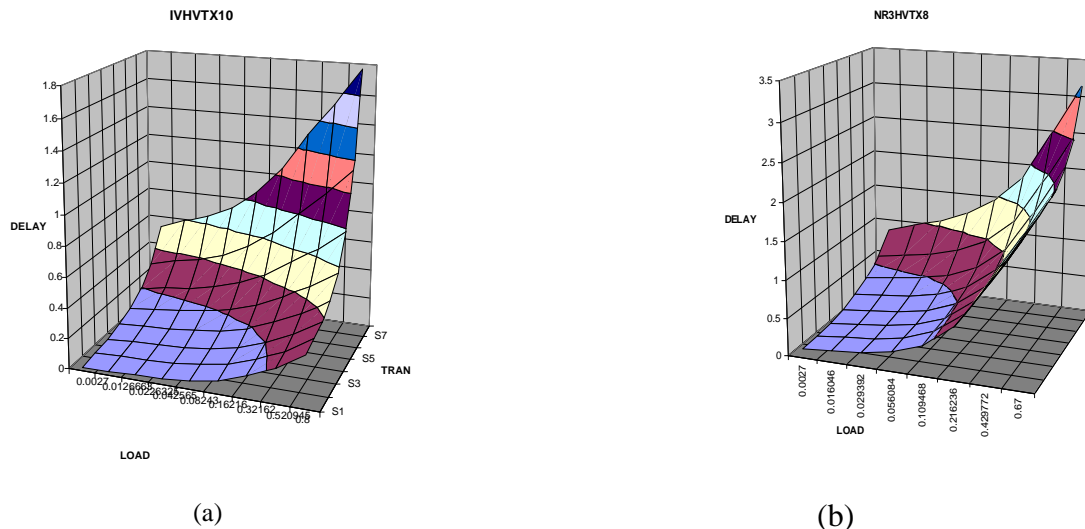


Figure 1

### 3. Placement Constrained (PLC) Algorithm

We have considered a segment of a path in a design, figure 2. Here three gates are connected to each other 1,2 and 3 with the interconnect load as L1 , L2 and L3. The delay of cell and the associated loads of 1,2 and 3 is D1 , D2 and D3 respectively.

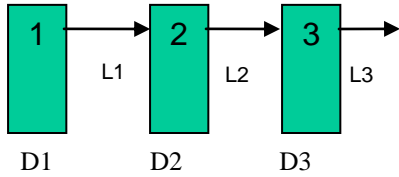


Figure 2

The cumulative delay of the segment is  $D1 + D2 + D3$ . Now let us shift the cell2 towards right such that the load between cell1 and cell 2 becomes  $(L1 + \partial x)$  and load between cell 2 and 3 is  $(L2 - \partial x)$  as shown in figure 3

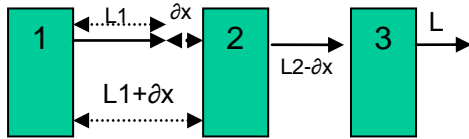


Figure 3

Now the cumulative delay of the segment will depend on  $\partial x$ . We have to find the value and direction on  $\partial x$  such that the cumulative delay is minimum.

#### 3.1 Polynomial approach to PLC Algorithm :

The delay and output transition of cells are function of input transition and output load. ,  $L_n$  is the load and  $IT_n$  is input transition for nth element.

$FD_n = fd_n \{(L_n), (IT_n)\}$  Delay function for Nth cell

$FOT_n = ft_n \{(L_n), (IT_n)\}$  Output transition function for Nth cell

For Cell 1

$FD1 = fd1 \{(L1), (IT1)\}$

$FOT1 = ft1 \{(L1), (IT1)\}$

We will now apply PLC algorithm here and displace cell2 towards right by  $\partial x$  as shown in fig 3. The load between cell 1 and cell 2 is  $(L1 + \partial x)$  and cell 2 and cell3  $(L2 - \partial x)$ .

For Cell1:

$FD1 = fd1 \{(L1 + \partial x), (IT1)\}$

$FOT1 = ft1 \{(L1 + \partial x), (IT1)\}$

For Cell2:

$FD2 = fd2 \{(L2 - \partial x), (ft1 \{(L1 + \partial x), (IT1)\})\}$

$FOT2 = ft2 \{(L2 - \partial x), (ft1 \{(L1 + \partial x), (IT1)\})\}$

For Cell3:

$FD3 = fd3 \{(L3), ft2 \{(L2 - \partial x), (ft1 \{(L1 + \partial x), (IT1)\})\})\}$

$FOT3 = ft3 \{(L3), ft2 \{(L2 - \partial x), (ft1 \{(L1 + \partial x), (IT1)\})\})\}$

The cumulative delay  $FD1 + FD2 + FD3 \dots$

is a function of  $\partial x$  where  $L1, L2, \dots, L_n$  are constants. In order to minimize total path delay we can differentiate sum of delays to calculate minima for  $\partial x$ .

Delay Function:  $FD = FD1 + FD2 + \dots + FD_n = f(\partial x, T1, L1, L2, \dots, L_n)$

Criteria to calculate  $\partial x$ :

$$\rightarrow \partial FD / \partial x = 0 \text{ and } \partial^2(FD) / \partial x^2 > 0$$

**Example of PLC algorithm:** There are various timing libraries used in the industry. One of them is NLDM in which delay and output transition is represented a function of input transition and output load in terms of lookup table. We can pick up few points from this table and derive an equation from the same. Lets consider the same segment again in figure 4.

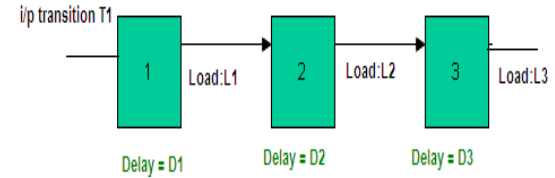


Figure 4

Delay for cell1 in terms of  $F1(x,y)$  where  $x$  is load,  $y$  is input transition and  $a1, b1, c1, d1$  are constants.

$$F1(x,y) = a1x + b1y + c1xy + d1$$

Output transition of cell1 in terms of  $P1(x,y)$  where  $x$  is load,  $y$  is input transition and  $g1, h1, i1, k1$  are constants.

$$P1(x,y) = g1x + h1y + i1xy + k1$$

Similarly for cell2 and cell3 the delay and output transition are :

Cell2:  $F2(x,y) = a2x + b2y + c2xy + d2$

$P2(x,y) = g2x + h2y + i2xy + k2$

Cell3:  $F3(x,y) = a3x + b3y + c3xy + d3$

$P3(x,y) = g3x + h3y + i3xy + k3$

Now lets shift cell2 towards left as in figure 5 such that new loads are

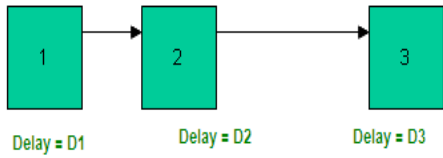


Figure 5

Now the delays and transition are :

For cell1:

$$\text{Delay} = D1(\partial x) = a1(L1 - \partial x) + b1T1 + c1(L1 - \partial x)T1 + d1$$

$$\text{O/P Transition} = T2(\partial x) = g1(L1 - \partial x) + h1T1 + i1(L1 - \partial x)T1 + k1$$

Similarly for cell2:

$$\text{Delay} = D2(\partial x) = a2(L2 + \partial x) + b2T2(\partial x) + c2(L2 + \partial x)T2(\partial x) + d2$$

$$\text{O/P Transition} = g2(L2 + \partial x) + h2T2(\partial x) + i2(L2 + \partial x)T2(\partial x) + k2$$

And for cell3 :

$$\text{Delay} = D3(\partial x) = a3L3 + b3T3(\partial x) + c3L3T3(\partial x) + d3$$

Substituting values of transition  $T2(\partial x)$  and  $T3(\partial x)$  in  $D2(\partial x)$  and  $D3(\partial x)$

$$D_{total} = D1(\partial x) + D2(\partial x) + D3(\partial x) = K1\partial x^2 + K2\partial x + k3$$

Where  $K1$ ,  $K2$  and  $K3$  are constants

$$K1 = c2(g1+i1) + (b3 + c3L3)[2i1(g1 + i1) - (g2 + h2(g1 + i1) - T2i2 + L2i1(g1+i1))]$$

$$K2 = a1 + c1T1 - a2 + b2(g1 + i1) + c2L2(g1 + i1) - c2T2$$

$$\text{Where } T2 = g1L1 + h1T1 + i1L1T1 + k1$$

Process for calculating the minima:

$$\frac{d D_{total}}{d \partial x} = 2K1 \partial x + K2 = 0$$

$$\frac{d \partial x}{\partial x} = \frac{-K2}{2K1}$$

$$\frac{d^2 D_{total}}{d^2 \partial x} = 2K1$$

$$d^2 \partial x$$

If  $2K1$  is positive then the point  $\partial x = -K2/(2*K1)$  is the required minima.

In this way we can calculate the loads on which the delay combination of three cells will be minimum.

### 3.2 Graphical representation of PLC algorithm:

In this section we will explain the algorithm graphically. Going back to the basic concept cells behavior vary from cell to cell. Lets us consider a timing critical segment of a path in figure 6.

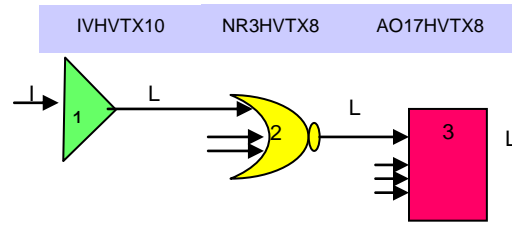


Figure 6

Fig 7a is the graph of delay vs input transition and output load and Fig 7b is the graph of output transition vs input transition and output load of cell1. Similarly Fig 8a and 8b is the graph of cell2.

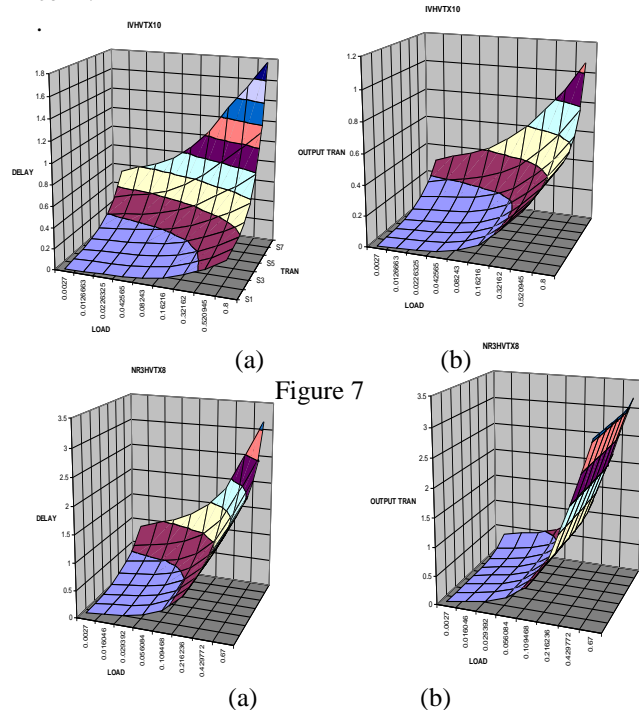


Figure 7

Figure 8

It is evident from the graphs in figure 7a and 8a that cell2, NR3HVTX8 is more sensitive to load than IVHVTX10. We will now apply the same algorithm to this segment and push the cell two towards right as in figure 9.

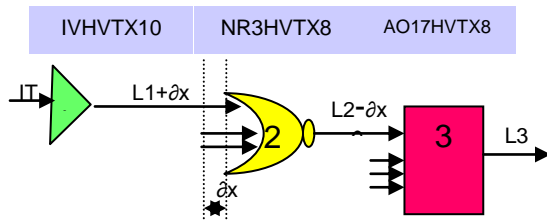


Figure 9

From the graph we can calculate the increment in delay and output transition. Similarly the decrement of delay and output transition of cell2 and cell 3 can be calculated from its delay and output transition graphs in Fig 7 and 8. Cumulative delta delay =  $\partial D1 - \partial D2 - \partial D3 < 0$ .

We can further increase  $\partial x$  to decrease cumulative delta delay till the point where it starts increasing. The point at which it just starts

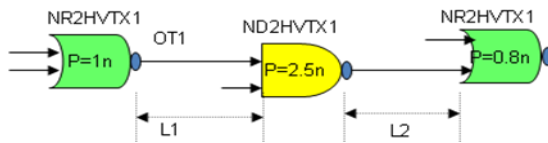


Figure 11

increasing is the minima. At this point the cumulative delay is minimum.

**Results:**

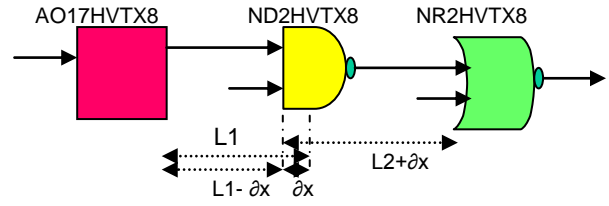


Figure 10

Consider a segment in Fig 10. The cell ND2HVTX8 is moved towards right. The individual delays and cumulative delays is plotted as a function of  $\partial x$  in figure 13 and 14. The cumulative delay has minima at the point B. The delay saving in the below segment in the below path was 11 %.

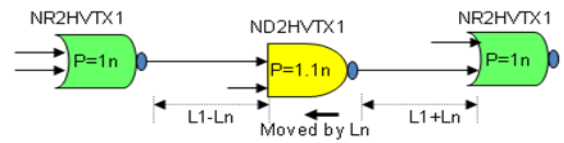


Figure 12

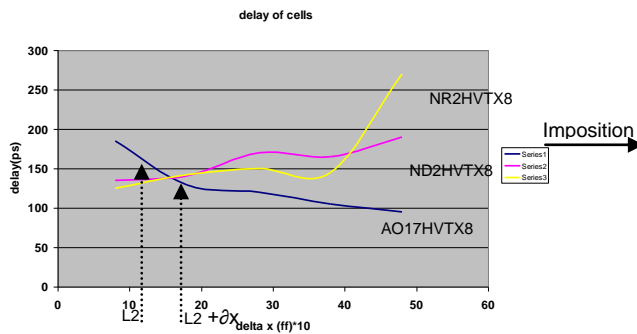


Figure 13

Power was also optimized on similar lines using polynomial and graphical approach. Figure 11 and 12 depicts power number optimization on similar lines using polynomial and graphical approach.

We have discussed only delay and power minima.

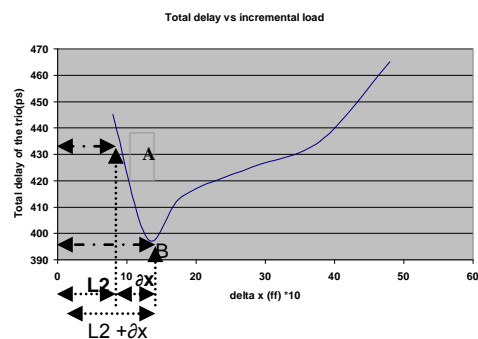


Figure 14

But the same concept can also be used for computing delay maxima for hold fixing during design closure.

**Conclusion:**

Polynomial and 3-D graphical approach is proposed for timing and power optimization. Based on results both timing and power have improved significantly. Thus presented PLC

algorithm finds extensive usage in low power and high speed designs.

**References:**

1. Andrew B Kahng, Stefanus Matik, Igor L. Markov, Min-max placement for large-scale timing optimization, International Symposium on Physical Design, Proceedings of the 2002 international symposium on Physical design, San Diego, CA, USA.

2. Singh K J, Wang . A. R, Brayton R. K, Sangiovanni –Vincentelli. A, Timing optimization of combinational logic, Computer-Aided Design, 1988. ICCAD-88. Digest of Technical Papers., IEEE International Conference, Santa Clara, CA, USA

3. Charles Alpert, Anirudh Devgan, Wire segmenting for improved buffer insertion, Annual ACM IEEE Design Automation Conference, Proceedings of the 34th annual conference on Design automation, Anaheim, California, United States