

Repeater Insertion combined with LGR Methodology for on-Chip Interconnect Timing Optimization

Michael Moreinis¹, Arkadiy Morgenshtein², Israel A. Wagner³ and Avinoam Kolodny⁴

1. Electrical Engineering Department, Technion, Haifa, Israel, E-mail: moreinis@tx.technion.ac.il
2. Electrical Engineering Department, Technion, Haifa, Israel, E-mail: arkadiy@tx.technion.ac.il
3. IBM Haifa Labs, Haifa University, Mount Carmel, Haifa, Israel, E-mail: wagner@il.ibm.com
4. Electrical Engineering Department, Technion, Haifa, Israel, E-mail: kolodny@ee.technion.ac.il

Abstract – Combination of Repeater Insertion with novel LGR (Logic Gates as Repeater) technique is presented, providing a methodology for delay optimization of CMOS logic circuits with RC interconnects. The traditional interconnect segmentation by insertion of repeaters is generalized to segmentation by distributing logic gates over interconnect lines and adding a reduced number of repeaters. Expressions for optimal segment length, optimal number of additional repeaters and scaling factors for both gates and repeaters are derived. An iterative solution is presented. Optimization results for several circuits are presented, showing significant improvement in performance in comparison with traditional repeater insertion.

I. Introduction

Interconnect optimization has become a major design consideration in state-of-the-art nanometer CMOS VLSI systems. The growth of die size together with decreased line width make wire delay more significant, compared with the active devices delay. For resistive wires, propagation delay increases quadratically with interconnect length because both capacitance and resistance of the interconnect increase linearly with length. In order to handle resistive interconnect, post-routing design steps have been added, involving wire segmentation and repeater insertion (Figure 1b) such that every segment resistance is much smaller than the on-resistance of the driver [2][5]. Wire sizing and gate sizing have also been applied at this stage [7][10].

Numerous studies explored various facets of the repeater insertion problem [4][6][8][9][10][11], adding inverters or buffers (double inverters) for amplifying logic signals on resistive wires between stages in a logic path. Besides speed optimization, this amplification reduces noise and restores logic levels [9]. However, the usage of repeaters implies a significant cost in power and area, without contributing to the logical computation performed by the circuit. A recent study [14] claims that in the near future, up to 40% of chip area will be used by inverters operating as repeaters and buffers. The use of numerous logically-redundant repeaters seems to be a waste, because the logic gates themselves may function as repeaters due to their amplifying nature. LGR (Logic Gates as Repeaters) [12] is a method of distributing logic gates over

interconnect; thus driving the partitioned interconnect without adding inverters to serve as repeaters (Figure 1c). In this paper a combination of LGR with traditional Repeater Insertion is presented (Figure 1d), where some segments are driven by logic gates and some by repeaters. The methodology exploits the potential of existing logic gates as repeaters, and only when this potential is fully utilized, additional repeaters are inserted. This technique may be called RI&LGR (Repeater Insertion combined with LGR).

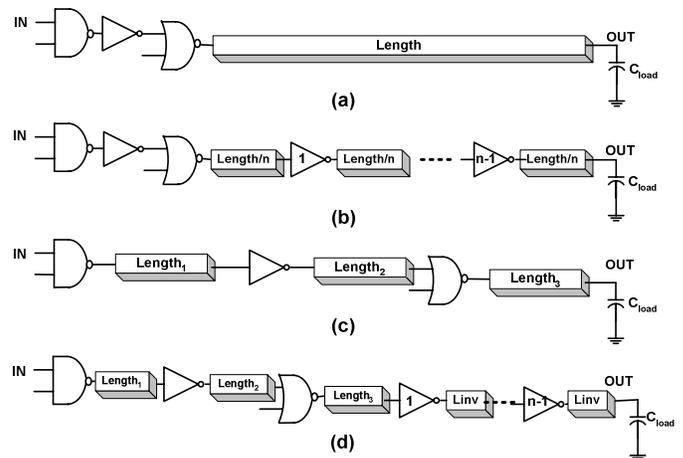


Figure 1. (a) A logic path driving a long interconnect wire. (b) Repeater insertion on the long interconnect (c) Logic gates are distributed over the interconnect and serve as repeaters (d) Logic gates and Repeaters are distributed over the interconnect.

II. Delay Formulation

The process of interconnect segmentation is schematically shown in Figure 2. Before the segmentation, logic gates are concentrated in a single logic block driving a long interconnect load (Figure 2a). After segmentation (Figure 2b) each gate/repeater drives assigned interconnect segment and the delay of each pair of logic-interconnect segment can be calculated separately. The overall delay is the sum of delays of all the combined logic-interconnect segments. In practice, the logic path can be laid out with wire segments in both x and y directions. The basic concept of matching wire segment lengths to their driver gates is the same.

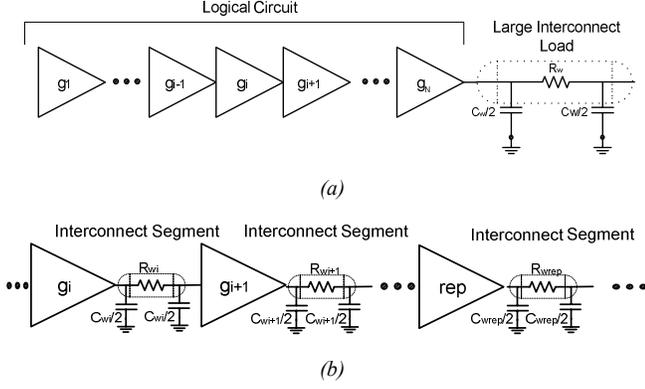


Figure 2. Logic gates with related interconnect load: (a) before segmentation, (b) after segmentation with repeaters insertion.

The delay modeling is similar to [12] and combines Logical Effort method [1] for gate/repeater calculation and Elmore delay model [3] for interconnect delay. For the combined i^{th} gate-interconnect segment in Figure 2b the respective delay components are:

$$D_{gate} = \tau \cdot \left(g_i \cdot \left(\frac{C_{i+1} + C_{w_i}}{C_i} \right) + p_i \right) \quad (1)$$

$$D_{interconnect} = R_{w_i} \cdot (0.5 \cdot C_{w_i} + C_{i+1})$$

where $\tau = R_{inv}C_{inv}$ is a technology-dependent time constant, defined as the delay of an ideal inverter driving another identical inverter. R_{inv} and C_{inv} are effective “on”-resistance and input capacitance of an inverter, respectively. Parameter p_i represents the parasitic delay of the gate and is related to capacitance of source/drain regions within the gate. C_i and C_{i+1} are input capacitance of gates i and $i+1$ respectively. C_{w_i} and R_{w_i} are the wire capacitance and resistance of segment i and can be replaced by:

$$C_{w_i} = L_i \cdot C_{int}, \quad R_{w_i} = L_i \cdot R_{int} \quad (2)$$

L_i is the length of the wire segment, C_{int} and R_{int} are the capacitance and resistance per unit length, respectively.

Repeaters are distributed uniformly and we define K as the number of additional repeaters and L_{inv} as the wire length assigned to each repeater. In order to account for the effect of logic, that drives the circuit being optimized, R_0 is defined as the driver output resistance. N is the number of gates, g_{inv} is the logical effort of the repeater ($=1$), p_{inv} is a repeater parasitic effort, and C_{load} is the load capacitance at the output of the circuit. Additional performance improvement can be gained by resizing gates and repeaters, where s_i is sizing factor of gate i and s_{inv} is repeaters scaling factor. The overall delay for the logic path is:

$$\begin{aligned}
D_{tot} = & R_0 \cdot s_1 \cdot C_1 + \\
& \sum_{i=1}^{N-1} \left[\tau \cdot \left(g_i \cdot \left(\frac{s_{i+1}C_{i+1} + L_i C_{int}}{s_i C_i} \right) + p_i \right) + \right. \\
& \left. + \left(0.5 \cdot L_i^2 R_{int} C_{int} + L_i R_{int} s_{i+1} C_{i+1} \right) \right] + \\
& \left[\tau \cdot \left(g_N \cdot \left(\frac{s_{inv} C_{inv} + L_N C_{int}}{s_N C_N} \right) + p_N \right) + \right. \\
& \left. + \left(0.5 \cdot L_N^2 R_{int} C_{int} + L_N R_{int} s_{inv} C_{inv} \right) \right] + \\
& \left[\tau \cdot \left(g_{inv} \cdot \left(\frac{s_{inv} C_{inv} + L_{inv} C_{int}}{s_{inv} C_{inv}} \right) + p_{inv} \right) + \right. \\
& \left. + \left(0.5 \cdot L_{inv}^2 R_{int} C_{int} + L_{inv} R_{int} s_{inv} C_{inv} \right) \right] + \\
& \left[\tau \cdot \left(g_{inv} \cdot \left(\frac{C_{load} + L_{inv} C_{int}}{s_{inv} C_{inv}} \right) + p_{inv} \right) + \right. \\
& \left. + \left(0.5 \cdot L_{inv}^2 R_{int} C_{int} + L_{inv} R_{int} C_{load} \right) \right] \quad (3)
\end{aligned}$$

The closed-form expression (3) provides a basis for analysis and timing optimization of a critical logic path involving long-distance wiring, using Logic Gates as Repeaters (LGR) and Repeater Insertion combined technique.

III. Optimization Method

In expression (3), there are $2N+2$ optimization parameters: L_i – wire length assigned to gate i , s_i – sizing factor of gate i , L_{inv} – wire length assigned to each repeater and s_{inv} – sizing factor of each repeater. We assume that each repeater drives the same wire length and is scaled by the same sizing factor. The total wire length L_{tot} is fixed and is given by summation of all assigned wire lengths, thus parameter K in expression (3) can be replaced by the following relations:

$$L_{tot} = \sum_{i=1}^N L_i + K \cdot L_{inv} \Rightarrow K = \left(L_{tot} - \sum_{i=1}^N L_i \right) / L_{inv} \quad (4)$$

The optimal solution can be obtained by applying (4) on (3) and differentiating the result with respect to $2N+2$ optimization parameters:

$$\frac{\partial D_{tot}}{\partial L_i} = 0, \quad \frac{\partial D_{tot}}{\partial s_i} = 0, \quad \frac{\partial D_{tot}}{\partial L_{inv}} = 0, \quad \frac{\partial D_{tot}}{\partial s_{inv}} = 0 \quad \forall i \in [1: N] \quad (5)$$

The solution yields the expressions in (6), that are interdependent. An iterative solution is proposed for solving the equations. The expressions in (6) are applied on initial set of parameters to get a set of new parameter values, and an iterative procedure is executed until convergence is obtained. An additional feature, required from the procedure is ability to determine the violations, ensuring $L_i \geq 0 \quad \forall i$, $L_{inv} \geq 0$.

$$\begin{aligned}
L_i &= \begin{cases} \frac{1}{L_{inv}} \cdot \left[\tau \cdot \left(g_{inv} \cdot \left(\frac{s_{inv} C_{inv} + L_{inv} C_{int}}{s_{inv} C_{inv}} \right) + p_{inv} \right) + (0.5 \cdot L_{inv}^2 R_{int} C_{int} + L_{inv} R_{int} s_{inv} C_{inv}) \right] - \tau \cdot g_i \left(\frac{C_{int}}{s_i C_i} \right) - R_{int} s_{i+1} C_{i+1} & 1 \leq i < N \\ \frac{1}{L_{inv}} \cdot \left[\tau \cdot \left(g_{inv} \cdot \left(\frac{s_{inv} C_{inv} + L_{inv} C_{int}}{s_{inv} C_{inv}} \right) + p_{inv} \right) + (0.5 \cdot L_{inv}^2 R_{int} C_{int} + L_{inv} R_{int} s_{inv} C_{inv}) \right] - \tau \cdot g_i \left(\frac{C_{int}}{s_i C_i} \right) - R_{int} s_{inv} C_{inv} & i = N \end{cases} \\
s_i &= \begin{cases} \sqrt{\frac{\tau \cdot (g_i \cdot (s_{i+1} C_{i+1} + L_i C_{int}))}{R_0 \cdot C_i \cdot C_i}} & i = 1 \\ \sqrt{\frac{g_i \cdot (s_{i+1} C_{i+1} + L_i C_{int})}{C_i \cdot \left(\frac{L_{i-1} R_{int} C_i}{\tau} + g_{i-1} \cdot \left(\frac{C_i}{s_{i-1} C_{i-1}} \right) \right)}} & 1 < i < N \\ \sqrt{\frac{g_i \cdot (s_{inv} C_{inv} + L_i C_{int})}{C_i \cdot \left(\frac{L_{i-1} R_{int} C_i}{\tau} + g_{i-1} \cdot \left(\frac{C_i}{s_{i-1} C_{i-1}} \right) \right)}} & i = N \end{cases} \\
L_{inv} &= \sqrt{\frac{\left(L_{tot} - \sum_{i=1}^N L_i \right) \tau \cdot g_{inv} \cdot (L_{inv} C_{int} + C_{load})}{C_{inv} \left(\tau \cdot g_N \cdot \left(\frac{L_{inv}}{s_N C_N} \right) + \left(L_{tot} - \sum_{i=1}^N L_i \right) R_{int} C_{inv} \right)}} \\
L_{inv} &= \sqrt{\frac{\left(L_{tot} - \sum_{i=1}^N L_i \right) (\tau \cdot g_{inv} + \tau \cdot p_{inv})}{0.5 \cdot R_{int} C_{int} \left(L_{tot} - \sum_{i=1}^N L_i \right) - R_{int} s_{inv} C_{inv} + R_{int} C_{load}} \quad (6)
\end{aligned}$$

IV. Results

In this Section RI&LGR optimization is demonstrated and compared with traditional Repeater Insertion. The Berkeley parameter extraction tool (BPTM) [13] was used to predict parameters of 0.07 μm process for both interconnect and device BSIM3v3 models.

A. Random Logic example

Optimization parameters, as presented in Section III, were obtained for random logic circuit in Figure 1 and the resulting delay was compared to timing improvement by traditional repeater insertion.

The test circuit contains 3 gates (NAND, NOT, NOR), that are initially in their minimal sizes. The interconnect is 1200 μm long, driver is 6 \times strength of minimal size inverter and load capacitance is 10 \times input capacitance of minimal inverter (10 \times C_{inv}). The optimal number of repeaters and their sizes are derived based on expressions similar to those presented in [2]. The un-optimized delay is 1.6 nsec, delay after repeater insertion is 0.42 nsec, and delay obtained by Iterative LGR after 6 iterations is 0.2 nsec. The optimal solution found by RI&LGR algorithm yields the following circuit configuration: L₁=0 μm , L₂=120 μm , L₃=160 μm , L_{inv}=480 μm , s₁= \times 16, s₂= \times 12, s₃= \times 25, s_{inv}= \times 48, K=2. This configuration, compared with traditional repeater insertion (K=5, s_{inv}=43), results in timing improvement of 52%. Both RI&LGR and Repeater Insertion achieve significant speed-up of 88% and 76% respectively, compared to un-optimized circuit.

It is interesting to analyze the solution behavior for various interconnect lengths. The delays of un-optimized circuit, circuit after repeater insertion and circuit after RI&LGR optimization, as a function of interconnect length are shown in Figure 3a. The un-optimized delay-vs.-length relation is quadratic, due to dependence of both capacitance and resistance on length. For both optimization techniques this relation is reduced to linear. For short wires, where a reduced number of repeaters is required, RI&LGR outperforms significantly the traditional repeaters. In this case no extra repeaters are added, thanks to logic gates acting as repeaters. For longer wires, the number of required repeater stages significantly exceeds the number of available logic stages, RI&LGR is still advantageous, though with less benefit, due to the fact, that major part of delay is spent on repeaters. Figure 3b presents the number of additional repeaters required for both optimization techniques. It is noticeable, that for any wire length, RI&LGR requires fewer repeaters and the difference is approximately the number of gates in the logic chain. In this example zero wire length was assigned to the first gate. Thus the first gate is not acting as a repeater (due to its low strength), and the RI&LGR saves only 2 repeaters.

B. A ripple-carry adder

The 4-bit ripple-carry adder in Figure 4 has a critical path which contains 17 CMOS gates (including the output inverters within OR and AND gates), and is indicated by the dashed line. The circuit drives 3000 μm interconnect. It was optimized by RI&LGR technique. The un-optimized delay is 5 nsec, delay after repeater insertion is 0.85 nsec, and delay

obtained by RI&LGR after 6 iterations is 0.62 nsec . RI&LGR outperforms the repeater insertion technique by 26%.

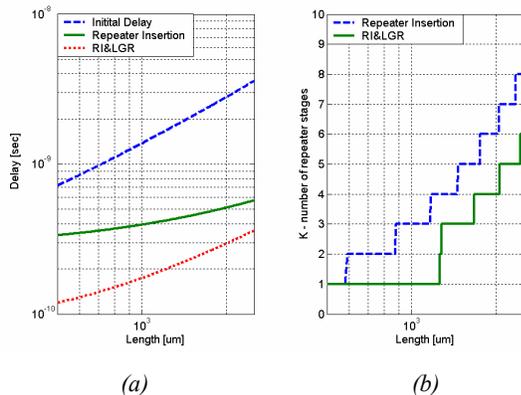


Figure 3. Algorithm behavior for various wire lengths for random logic circuit: (a) delay (b) number of additional repeaters.

Figure 5 presents the behavior of the algorithm applied on ripple carry adder for various wire lengths. The number of required repeaters grows linearly with the wire length. For RI&LGR the number of additional repeaters is unchanged until all the available logic gates are utilized as repeaters. For short interconnect RI&LGR outperforms significantly the traditional repeater insertion. For long interconnect, where the number of required repeater stages significantly exceeds the number of available logic stages, the advantage of RI&LGR is less noticeable.

V. Conclusions

Timing optimization methodology, where traditional repeater insertion is enhanced by distributing of logical gates over resistive interconnect has been presented. The logic gates thus serve also as repeaters, driving wire segments. The expressions for optimal wire length assignment to each gate, gates sizing factors, number of repeaters and repeaters scaling factor were obtained. An iterative solution was proposed to solve $2N+2$ equations. Several circuits were simulated and characterized using RI&LGR methodology. Design experiments indicate that RI&LGR can provide viable improvement to traditional Repeater Insertion for VLSI interconnect optimization.

VI. References

[1] I. Sutherland, B. Sproull, D. Harris, "Logical Effort - Designing Fast CMOS Circuits", Morgan Kaufmann Publishers, 1999
 [2] H.B. Bakoglu, "Circuits, Interconnections and Packaging for VLSI", Adison-Wesley, 1990, pp. 194-219.
 [3] W. C. Elmore, "The transient response of damped linear networks with particular regard to wide band amplifiers," *J. Appl. Phys.*, vol. 19, no. 1, 1948.
 [4] V. Adler and E. G. Friedman, "Uniform Repeater Insertion in RC Trees," *IEEE Tran. on Circuits and Systems I: Fundamental Theory and Applications*, Vol. 47, No. 10, pp. 1515-1523, October 2000.

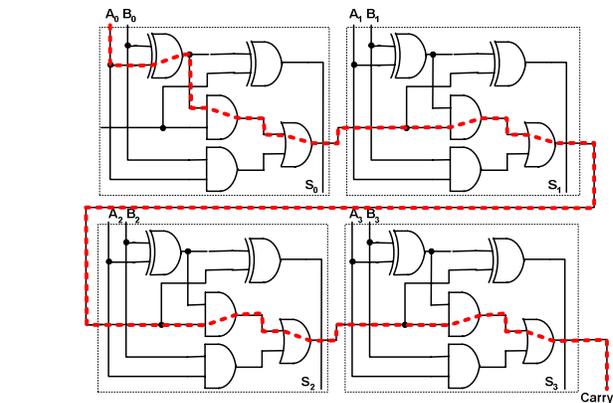


Figure 4. Ripple carry adder

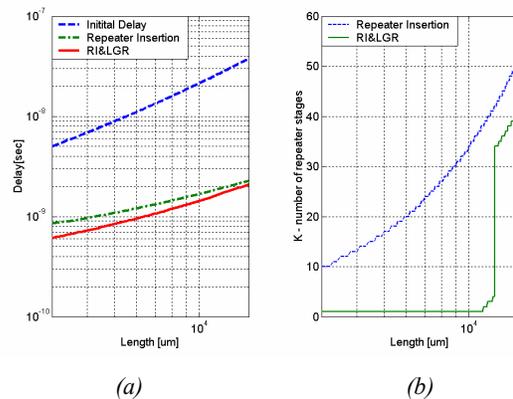


Figure 5. Algorithm behavior for various wire lengths for ripple carry adder (a) delay (b) number of additional repeaters.

[5] V. Adler and E. G. Friedman, "Repeater Design to Reduce Delay and Power in Resistive Interconnect," *IEEE Tran. on Circuits and Systems II: Analog and Digital Signal Processing*, Vol. CAS II-45, No. 5, pp. 607-616, May 1998.
 [6] L. V. Ginneken, "Buffer Placement in Distributed RC-tree Networks for Minimal Elmore Delay," *Proc. IEEE Int'l Symposium on Circuits and Systems*, pp. 865 - 868, May 1990.
 [7] J. Lillis, C. K. Cheng and T. T. Y. Lin, "Optimal Wire Sizing and Buffer Insertion for Low Power and a Generalized Delay Model," *Proc. IEEE Int'l. Conf. on Computer-Aided Design*, pp. 138-143, Nov. 1995.
 [8] C. J. Alpert and A. Devgan, "Wire segmenting for improved buffer insertion," in *Proc. Design Automation Conf.*, Anaheim, CA, June 1997
 [9] C.J. Alpert, A. Devgan, S. T. Quay, "Buffer Insertion for Noise and Delay Optimization," *Proc. 34th ACM/IEEE Design Automation Conference*, pp. 362-367, 1999.
 [10] C. Chu and D. F. Wong, "Closed Form Solution to Simultaneous Buffer Insertion / Sizing and Wire Sizing," *ACM Trans. on Design Automation of Electronic Systems*, vol. 6, no. 3, pp. 343-371, July 2001.
 [11] S. Dhar and M. A. Franklin, "Optimum buffer circuits for driving long uniform lines," *IEEE J. Solid-State Circuits*, vol. 26, pp. 32-40, Jan. 1991.
 [12] A. Morgenshtein, M. Moreinis, I. Wagner and A. Kolodny, "Logic Gates as Repeater (LGR) for Timing Optimization of SoC Interconnects," *Proc. IFIP Int'l Conf. on VLSI SOC*, Darmstadt, pp. 99-104, Dec. 2003.
 [13] Berkeley Predictive Technology Model (BPTM), www-device.eecs.berkeley.edu/~ptm/introduction.html.
 [14] J.A. Davis, R. Venkatesan, K. A. Bowman and J. D. Meindl, "Gigascale integration (GSI) interconnect limits and n-tier multilevel interconnect architectural solutions," *Proc. of the Int'l Workshop on System Level Interconnect Prediction (SLIP)*, San Diego, pp. 147-148, April 8-9 2000.