

Custom Layout Strategy for Rectangle-Shaped Log-Depth Multiplier Reduction Tree

Patrik Kimfors, Niklas Broman, Andreas Haraldsson, Kasyab P. Subramaniyan,
Magnus Sjölander, Henrik Eriksson, and Per Larsson-Edefors
VLSI Research Group, Dept. of Computer Science and Engineering,
Chalmers University of Technology, SE-412 96 Gothenburg, Sweden
{kasyab,perla}@chalmers.se

Abstract—Multiplier reduction trees that have a logarithmic logic depth generally exhibit poor regularity, in terms of how the gates are interconnected. Consequently, it is well known that Partial-Product Reduction Trees (PPRTs), such as Wallace, Dadda and TDM, are very difficult to lay out in a custom design flow. However, our previously proposed HPM scheme enabled the implementation of a PPRT that consistently uses regular interconnect patterns, without sacrificing the performance originating from the logarithmic logic depth. The original HPM layout is perfectly regular, however, it has a PPRT in the shape of a wide triangle. We now propose a custom layout strategy for the HPM scheme that leads to rectangle-shaped PPRTs that are more straightforward to reconcile with the need for a rectangular multiplier footprint. The proposed layout strategy is implemented and evaluated for a 16-bit 90-nm design.

I. INTRODUCTION

Among the elementary datapath units, the integer multiplier is large and, if not pipelined, restricts overall performance. To obtain low multiplier delay, Partial-Product Reduction Trees (PPRTs) are often based on algorithms, such as Wallace [1], Dadda [2] or TDM [3], which connect the partial-product-reducing gates in a manner that leads to a logic depth that is proportional to the logarithm of the input data word length. However, the log-depth PPRT algorithms mentioned above do not take layout aspects into account, but produce generic netlists that turn out to have their adder cells irregularly interconnected. This makes the ensuing layout phase very complicated, if a custom design flow is adopted.

Several types of multipliers that combine ease of design with high performance have previously been proposed. Hybrid multipliers, such as the overturned-stairs tree [4], represent a path to multiplier design that makes a compromise between speed and regularity. Furthermore, it has been shown that by deploying a number of speed-improving techniques to array-style multipliers, the performance can approach that of PPRT-based multipliers [5].

The High-Performance Multiplier (HPM) scheme [6], [7] is based on the Dadda algorithm, but has a different order in which adder cells are assigned to sums, carries, and other partial products: Carry and sum bits are compressed as late as possible. The result is a perfectly regular interconnect pattern between all adder cells inside the PPRT, and this makes the placement of cells as straightforward as for a slow array multiplier. Clearly the regularity of HPM can greatly simplify

custom layout, but it also simplifies the implementation of other features such as the twin-precision technique [8], which allows a multiplier to be reconfigured into different computational modes, after the ASIC has been fabricated.

Originally the HPM scheme was associated with a triangular layout, where each column of the triangle represented one level of product significance. This layout strategy was simple to explain and straightforward to route, however, it could lead to low area utilization during custom layout work, as the sides of the triangle may be hard to use entirely for the circuitry that drives the PPRT's primary partial-product inputs. The key contribution of this work, thus, is a new layout strategy that allows us to lay out a rectangle-shaped HPM PPRT.

II. RECTANGULAR HPM LAYOUT

A. The Original HPM Cell Organization

The original HPM layout strategy [7] targeted a triangle-shaped design to simplify routing. This strategy made routing simple in that all wiring for signals that are going to be reduced are routed vertically between or over cells, as shown in Fig. 1(a).

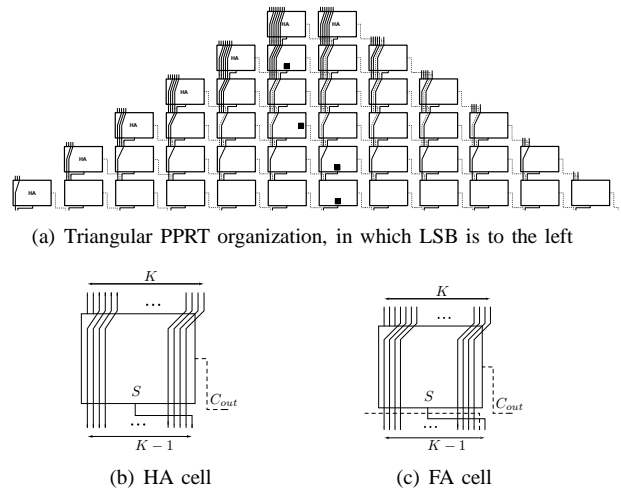


Fig. 1. Original cell organization for an 8-bit HPM reduction tree. The short wire stubs in (a) represent primary partial-product bits. The black squares in (a) illustrate the critical path.

At a more detailed level, Fig. 1(b) shows the over-the-cell routing patterns for the half adders: There are K partial-

product bits that come in at the top. The half-adder (HA) cell reduces the inbound signals by one and produces an output carry that is passed rightwards to the next column. The half adder, which operates on the leftmost two partial-product bits, places the sum bit as far to the right as possible among the $K - 1$ outputs. The full-adder (FA) cells are governed by a similar strategy: We process the sum bit as late as possible, as before, but we also process the input carry as late as possible, as shown in Fig. 1(c). Thus, these two bits are passed as far to the right as possible among the $K - 1$ outputs.

B. Turning an HPM Triangle into an HPM Rectangle

With the motivation that a rectangular organization of the PPRT cells would yield a higher PPRT area utilization, an HPM PPRT in the shape of a rectangle was conceived. The number of adder cells remains constant regardless of layout strategy, so in order to make the PPRT rectangular we "only" needed to rearrange how the cells are placed. It turns out that all adder cells can be moved around in a regular pattern. Fig. 2 shows how some of the cells from the lower rows of a 16-bit HPM PPRT design were rearranged.

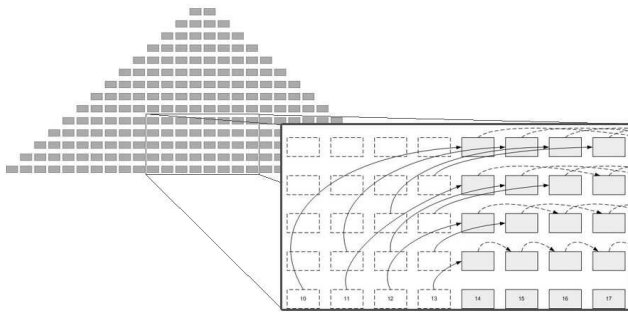


Fig. 2. The transformation used to change the HPM PPRT cell organization from triangular to rectangular.

By rearranging the placement of the adder cells the signal propagation paths are moved accordingly. Fig. 3 highlights the critical path of a triangular and rectangular PPRT, respectively. Although the exact increase in distance between cell pins on the critical path will depend on the cell aspect ratio, the figure suggests that the rectangular PPRT will have longer critical wires and, consequently, longer delay. Because 45° design rules are stricter on minimal spacing than Manhattan rules, routing congestion is more likely in the rectangular PPRT.

C. Evaluation of 16-bit HPM Layouts

Both a rectangular and a triangular PPRT were laid out in a 90-nm process technology, see Fig. 4 and 5. This work indeed focuses on the PPRT structure, however, to properly account for the impact on delay and power dissipation of the PPRT's primary partial-product inputs, Partial-Product Generation (PPG) circuitry is needed. Made up of 2-input AND gates, the PPG circuits of this evaluation target unsigned 16-bit input operands. For layout simplicity, the PPG circuitry has been placed outside the core of the PPRT layouts.

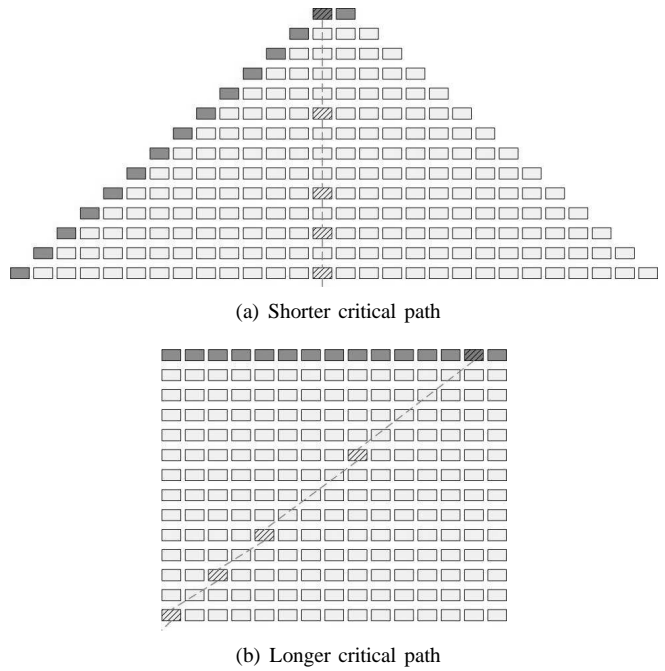


Fig. 3. The critical path of (a) a triangularly organized PPRT, and (b) a rectangularly organized PPRT.

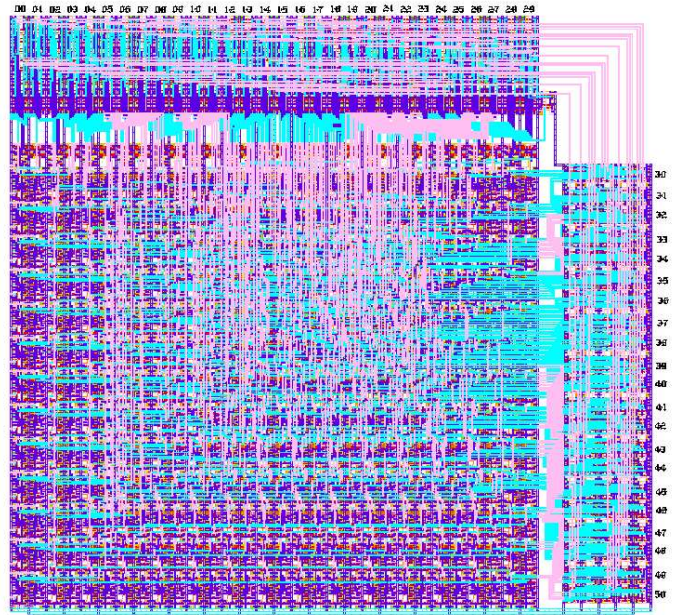


Fig. 4. Rectangular 16-bit HPM PPRT with PPG circuitry.

The PPRT rectangle had a rectangular footprint of $6,500 \mu\text{m}^2$ with an almost 100% area utilization. In contrast, the PPRT triangle spanned a rectangular area of $11,500 \mu\text{m}^2$, but with a much lower area utilization. Based on layout extracted RC netlists, we analyzed timing and power properties: The triangular and the rectangular layout had a worst path delay

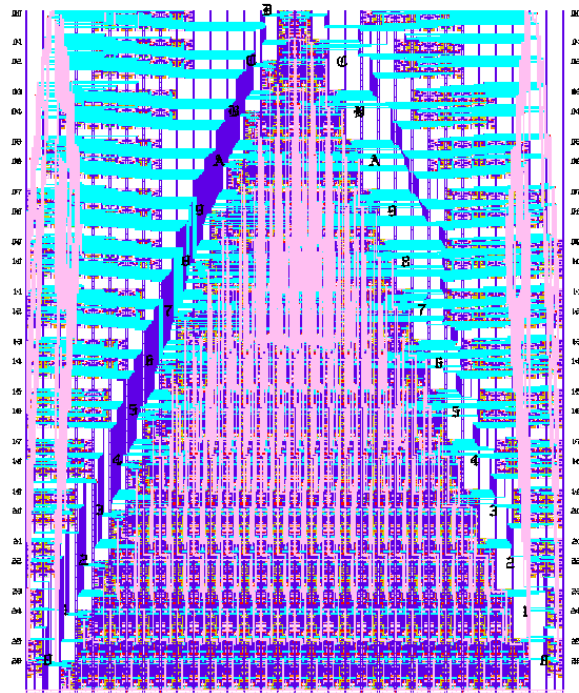


Fig. 5. Triangular 16-bit HPM PPRT with PPG circuitry.

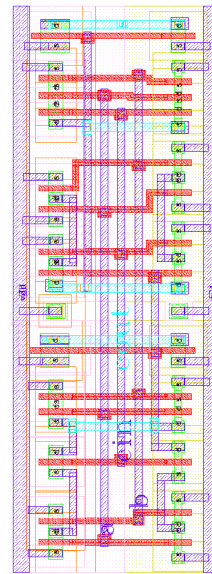
(reported by Synopsys PathMill) of 1.60 ns and 1.69 ns, respectively, when considering PPRT and PPG circuitry. The power dissipation was 1.17 mW and 0.99 mW for the triangular and the rectangular layout, respectively. These values were obtained from an Hspice simulation over 100 pseudo-random vectors at a 500-MHz clock rate and a 1-V supply voltage, considering both the PPRT and PPG circuitry.

The difference in critical path delay was expected, since the rectangular layout tends to increase wire lengths. The power dissipation difference is more of a surprise and can only be explained by differences in glitching¹; the path delays tend to even out in the rectangular layout, suppressing many of the glitches that exist in the triangle. Although no complete analysis has been made regarding the exact reason for the power reduction, a quick inspection of the waveforms of a few nodes of the two designs confirmed that glitches appear to be consistently fewer in the rectangular HPM.

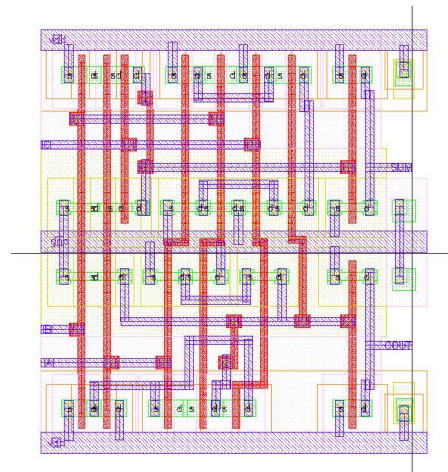
D. Adder Cell Design

The FA and HA cells used in the designs were custom made. As an example of cell layout, the FA cell of the triangular and the rectangular PPRT is shown in Fig. 6(a) and Fig. 6(b), respectively. The transistor schematics are identical, however, to obtain a favorable overall PPRT aspect ratio, the cells are differently organized.

¹A glitch is the useless signal transition on a gate output that may result from inputs arriving at different points in time. Especially slow array multipliers are infamous for their large amount of glitch power [9].



(a) Triangle-PPRT FA cell



(b) Rectangle-PPRT FA cell

Fig. 6. Full-Adder (FA) cell layouts.

The custom layout effort on the FA and HA cells was limited. Furthermore, since the optimal cell aspect ratio was an issue that was under investigation, no layout compaction was carried out, leading to a 75% area overhead for the FA cells of the rectangular PPRT, when compared to the nominal-strength standard cell of the vendor library.

E. Scaling to Larger Word Lengths

In the design of the 16-bits rectangular HPM PPRT, the minimum spacing was not always used during routing. This means that there is space for an additional signal with a slightly different routing, and that it is not necessary to double the size of adder cell area, to avoid using more metal layers when the word length is increased to 32 bits. An additional consequence of increasing cell sizes is that it is possible, in our design, to

make a channel for signals in metal 1 that relaxes the need to increase the cell width as much as the cell height.

TABLE I
APPROXIMATE ASPECT RATIO FOR RECTANGULAR PPRT.

Word length	Max #signals	PPRT Height/width
16	10	0.90
32	21	1.35
64	42	2.01

Assuming the use of a signal channel in metal 1, Table I presents an estimation of PPRT aspect ratios for increased word lengths.

F. Layout Strategy

Considering the experiences from the layout work in Sec. II-C, it is now time to describe a systematic placement and routing strategy for the rectangular HPM PPRT. As shown in Fig. 7, each cell of the rectangle consumes the signals generated by cells above, starting from the bottom wire. Conversely, wires are added in the upper part of the cell. Preferably a Manhattan-style layout with fine-grain jogs is used to allow for wire crossings. The routing of the designs in Sec. II-C was using coarser jogs and relatively many vias.

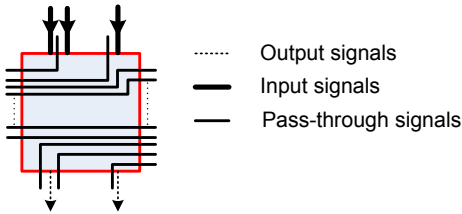


Fig. 7. General routing scheme per cell of a rectangular HPM PPRT.

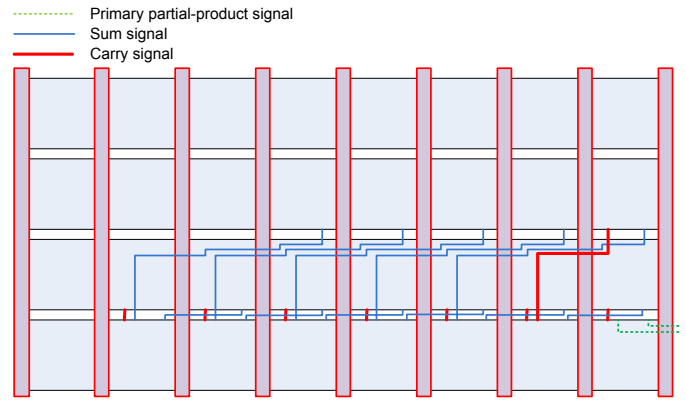
For PPRTs that support larger word lengths, dedicated wire channels that carry longer vertical wires in lower metal layer may prove useful. However, the layout strategy of Fig. 7 would generally remain the same.

As an example, Fig. 8(a) shows how the routing is handled for the bottom three rows of the PPRT rectangle, while Fig. 8(b) shows the next routing phase, when one more row is considered. Here we assume that the primary partial-product signals are inserted from the right side of the rectangle.

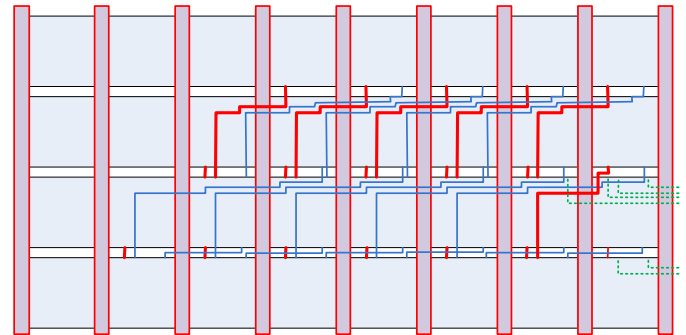
III. CONCLUSION

The fastest multipliers are based on logarithmic logic depth gate circuits, which are intrinsically hard to layout since they are irregularly interconnected. The original HPM scheme managed to combine log-depth circuitry with a regular interconnect pattern, but the PPRT layout was triangularly shaped and during custom layout it was hard to reconcile this with the need for a rectangular footprint of the complete multiplier block.

We have presented a custom layout strategy for rectangle-shaped PPRTs that can simplify the layout implementation of efficient HPM multipliers. To evaluate the rectangular PPRT two 16-bit 90-nm PPRTs were laid out; one rectangular and



(a) Routing scheme for rows $N - 4$, $N - 3$ and $N - 2$



(b) Routing scheme for rows $N - 5$, $N - 4$ and $N - 3$

Fig. 8. Routing in a rectangular HPM PPRT that handles the multiplication of two N -bit operands.

one triangular, for reference. The rectangular PPRT turned out to be 5.6% slower than the triangular PPRT. However, the power dissipation of the rectangular PPRT is 15.4% lower, for the same clock rate.

REFERENCES

- [1] C. S. Wallace, "A Suggestion for a Fast Multiplier," *IEEE Transactions on Electronic Computers*, vol. 13, pp. 14–17, Feb. 1964.
- [2] L. Dadda, "Some Schemes for Parallel Adders," *Alta Frequenza*, vol. 34, no. 5, pp. 349–356, May 1965.
- [3] V. G. Oklobdzija, D. Villegier, and S. S. Liu, "A Method for Speed Optimized Partial Product Reduction and Generation of Fast Parallel Multipliers Using an Algorithmic Approach," *IEEE Transactions on Computers*, vol. 45, no. 3, pp. 294–306, Mar. 1996.
- [4] Z.-J. Mou and F. Jutand, "'Overturned-Stairs' Adder Trees and Multiplier Design," *IEEE Transactions on Computers*, vol. 41, no. 8, pp. 940–948, Aug. 1992.
- [5] Z. Huang and M. D. Ercegovic, "High-Performance Low-Power Left-to-Right Array Multiplier Design," *IEEE Transactions on Computers*, vol. 54, no. 3, pp. 272–283, Mar. 2005.
- [6] H. Eriksson, "Efficient Implementation and Analysis of CMOS Arithmetic Circuits," Ph.D. Thesis, Chalmers University of Technology, 2003.
- [7] H. Eriksson, P. Larsson-Edefors, M. Sheeran, M. Sjalander, D. Johansson, and M. Schölin, "Multiplier Reduction Tree with Logarithmic Logic Depth and Regular Connectivity," in *IEEE International Symposium on Circuits and Systems*, 2006, pp. 5–8.
- [8] M. Sjalander, H. Eriksson, and P. Larsson-Edefors, "An Efficient Twin-Precision Multiplier," in *Proc. of IEEE Intl Conf. on Computer Design*, October 2004, pp. 30–33.
- [9] T. K. Callaway and E. E. J. Swartzlander, "Optimizing Multipliers for WSI," in *Proceedings of the Fifth Annual IEEE International Conference on Wafer Scale Integration*, 1993, pp. 85–94.