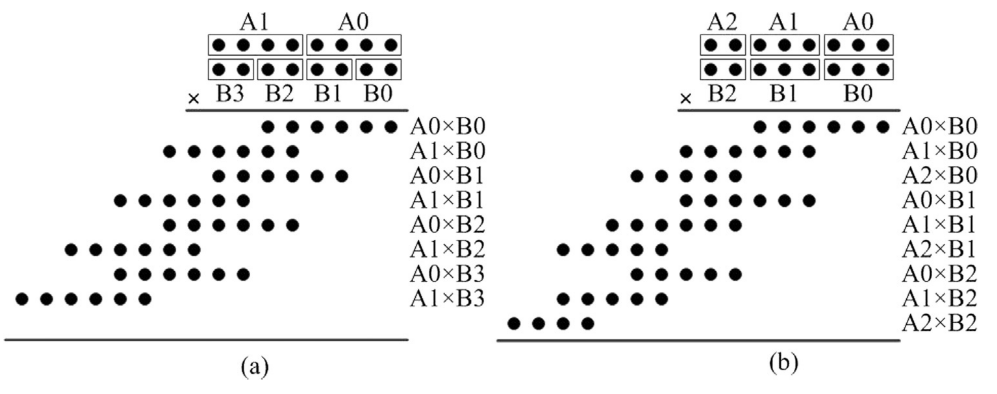
**Fast Multiplier Generator for FPGAs with LUT based Partial Product Generation and Column/row Compression**

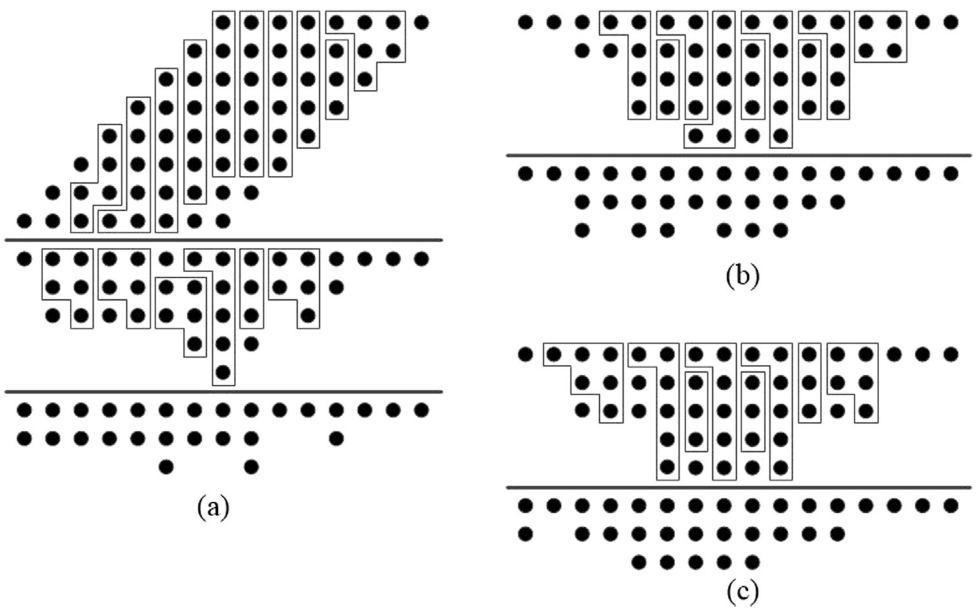
**——INTEGRATION, the VLSI journal**

1. **Contributions**
   1. Propose a combo multiplier generator that improves and combines the best practices in multiple works in the previous literature, include:
      1. GPC based CCT
      2. PPG with 4x2 and 3x3 multipliers
      3. Final adder width minimization during CCT generation (i.e., row compression)
      4. Ternary final adder to utilize FPGA's carry-chains
   2. Proposed method packs the native LUTs of the FPGA both in the PPG stage and the CCT stage
2. **Partial Product Generation (PPG)**
   1. Our PPG creates bigger but fewer partial products by implementing small 4×2 and 3×3 multipliers using the native 6-input LUTs of Xilinx FPGAs. (Notice that 4+2=3+3= 6.)



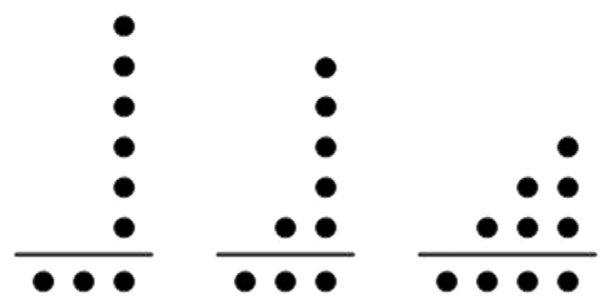
8x8 multipliers with (a )4x2 and (b) 3x3 PPG

* 1. Each 4×2 PPG multiplier generates one of the 8 partial products and consumes 6 LUTs as each multiplier has 6 bits of output, while each LUT has only one bit of output.
  2. Save 1~2 CCT levels(height of stacked bits after partial product generation)



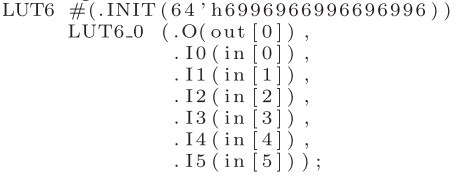
CCT with (a) conventional (b) 4x2 PPG (c) 3x3 PPG

1. **Generalized Parallel Counters (GPCs)**
   1. Implement CCT with GPCs, use GPC(a, b, c; z) to describe a GPC, where a, b, c are the number of input bits of rank2, rank1, rank0(from left to right), z is the number of output bits of the GPC.

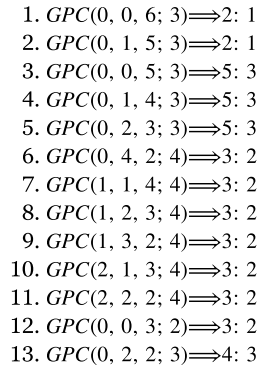


(a) GPC(0,0,6;3) (b) GPC(0,1,5;3) (c) GPC(1,2,3;4)

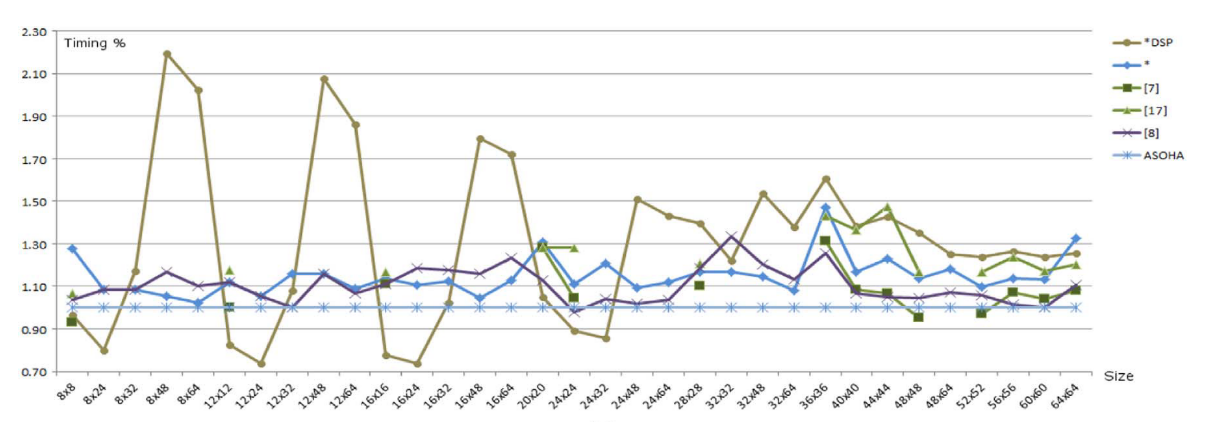
* 1. Implement GPCs by instantiating the LUT6 Xilinx primitive. The number of instantiations depends on the number of output bits of the GPC.



1. **Column Compression Tree (CCT)**
   1. The CCT of our implementation is based on GPCs as shown by the example above. GPCs have 6 input bits and 4 output bits at most.
   2. List below is a priority list, the GPC at the top is the most preferred.



1. **Results**



Normalized timing results, ASOHA is proposed method in this paper