# Results

To evaluate the performance of STT-LUT implementation, two main methods are used: individual gate mapping and multiple gate mapping. Through netlisting in HSPICE, results of the delay, active power (both at minimum period, which is 26 ps, and at a 1ns period), the power delay product (PDP), the stand-by power, and the area of the adder can be obtained. To evaluate the effectiveness of STT-LUT implementation, there must be a comparison of the results in which the adder’s gates are not reconfigured. This is called an all custom CMOS. The critical output, which is the indication of where the critical path ends and if the critical path has changed at all, was observed.

## Single Gate Implementation Results

For individual gates mapping, the majority of the gates were replaced with LUT2, NOT gates N1 and C0B were replaced with LUT1, NOR gate C2 was replaced with LUT3, AND gate C3 was replaced with LUT4, Gate C4, which is a combination of two AND gates and a NOR gate, was also replaced with LUT4. Though this gate apepars to have 5 inputs, there are only 4 unique inputs due to the output of gate GB3 becoming an input to each of the two AND gates. A visual is shown in figure 5.

|  |  |
| --- | --- |
| Figure 5. Various LUT mappings. | Figure 6. Gates with no delay overhead. |

As expected from STT-LUT characteristics, STT-LUT replacement of a gate may result in delay overhead. The minimum period is the period when none of the gates are replaced with LUTs, which is the all custom CMOS and it is observed to be 26 ps. The identified gates that do not produce overhead are shown in figure 6. As observed from the results, all gates in the critical path and some gates that are not in the critical path (referred to as non-critical gates) cause delay. The individual gate mapping results are shown in a bar graph at figure 7.

The mapping of critical gate C3 produced the highest delay of 34 ps, which is 1.31 times greater than the delay of the all custom CMOS. Another critical gate, N2, produced the second highest delay of 32 ps, which is 1.23 times greater than the all custom CMOS delay. The mapping of a non-critical gate, such as C4, displayed the third highest delay of 29 ps, which is 1.15 times greater than the all custom CMOS delay. Gate C4 resulted in the same minimum period as the critical gate S3.

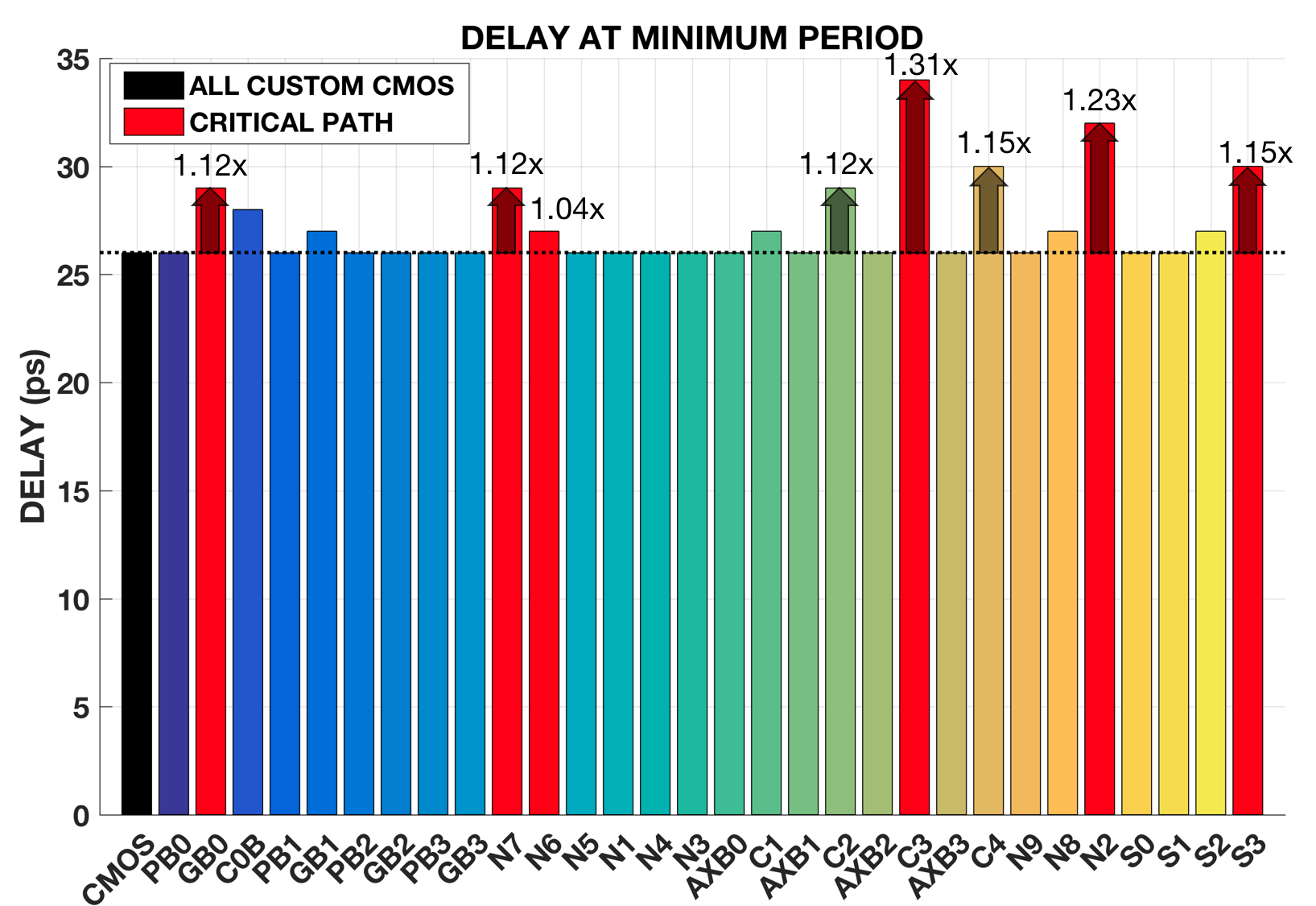


Figure 7. Delay at minimum period (individual mapping). The black bar represents the all custom CMOS. The red bars represent gates along the critical path. The multicolored bars represent the non-critical gates.

The active power is measured in two different time periods: at its minimum period and at a 1 ns period. Comparison of the active power at the 1 ns period is more reasonable because the minimum period of each gate may be different from other gates.

As expected from STT-LUT characteristics, STT-LUT replacement of any gate results in active power, area, and PDP overhead. LUTs have a larger area, which is due to more complex design of multiplexers and transistors, than logic gates. So the higher n-input gate, there is a consequence of larger area thus more active power is required. The PDP is the product of the delay and active power at the minimum period of each gate’s LUT implementation. PDP is the energy consumed by the adder when the gates are mapped with LUTs. The results of active power, area, and PDP are shown in a set of bar graphs at figures 8, 9, and 10, respectively.

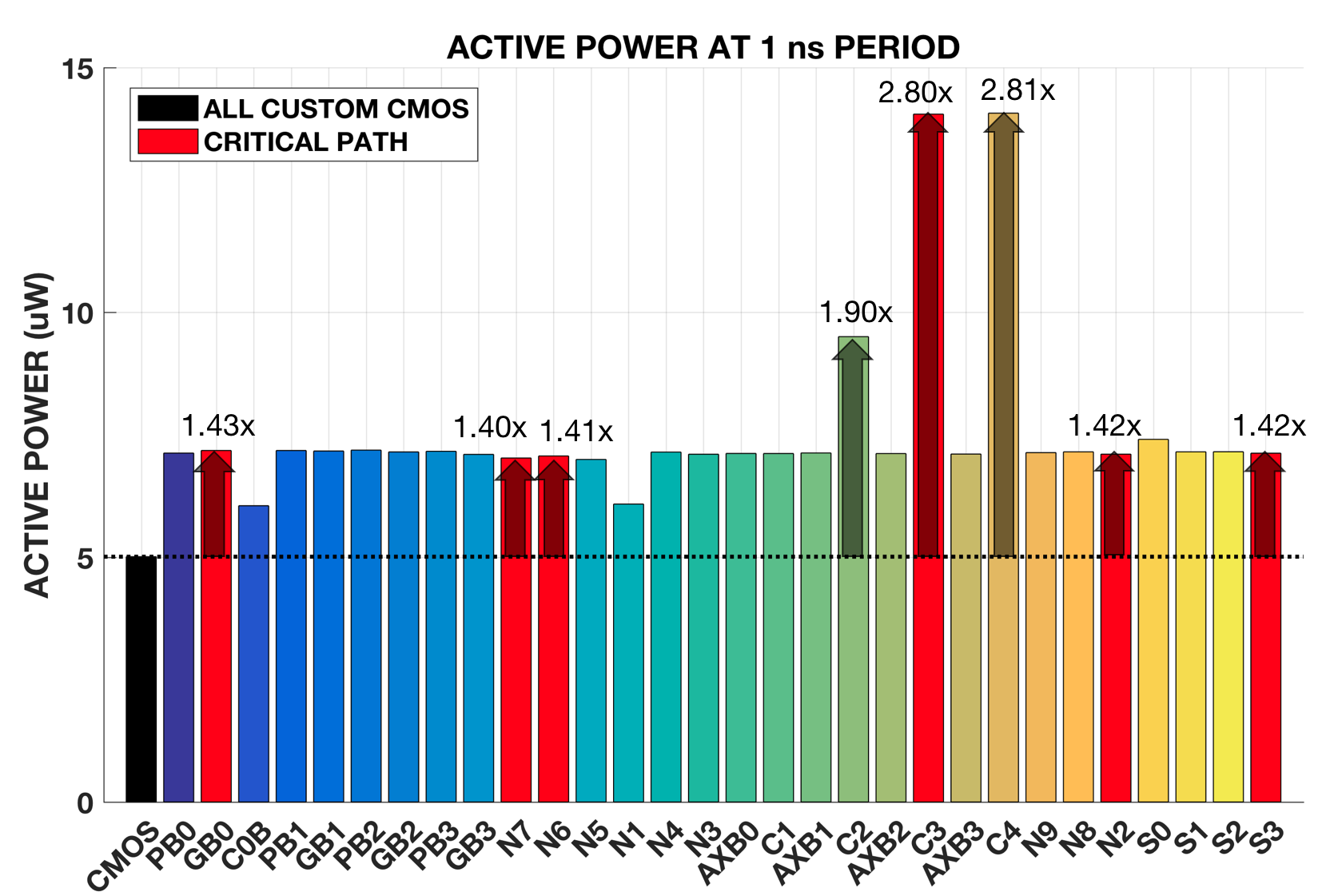


Figure 8. Active power at a 1 ns period (individual mapping).

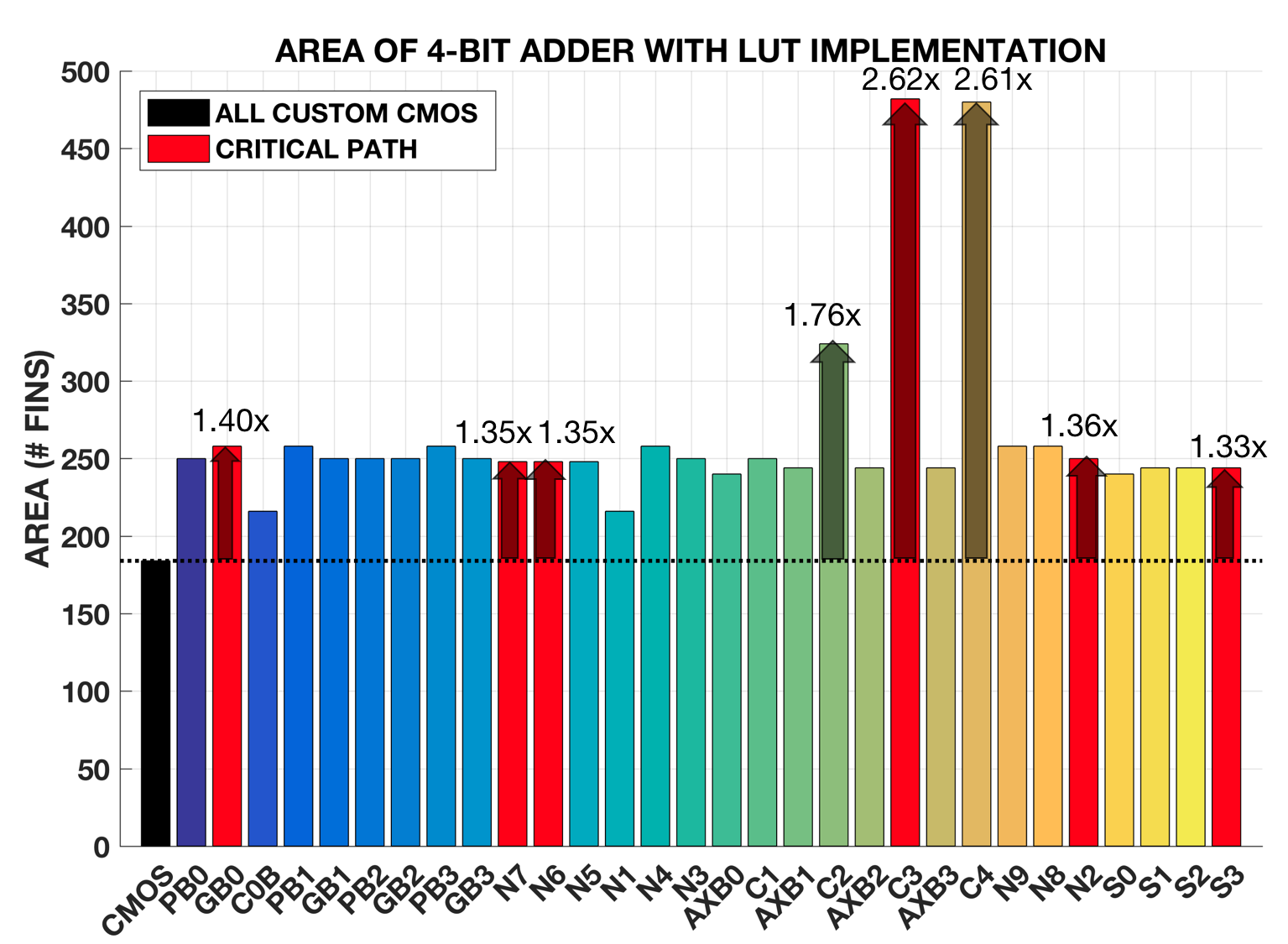


Figure 9. Area of 4-bit adder (individual mapping).

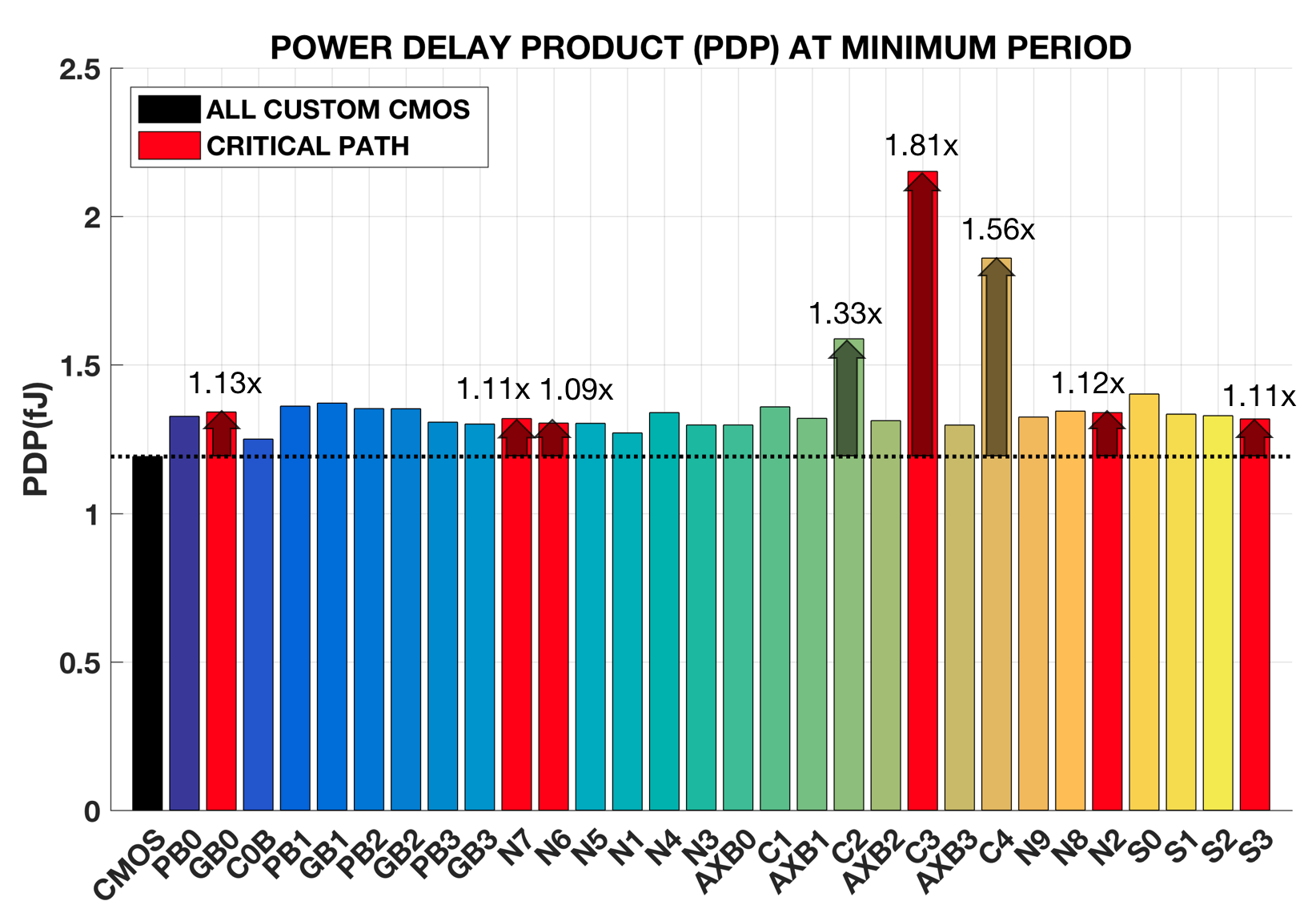


Figure 10. PDP at minimum period (individual mapping)

The mapping of gates C3 and C4 resulted in an active power 2.80 and 2.81 times greater than the all custom CMOS and their area overhead are 2.62 and 2.61 times greater. When gate C4 is implemented, it has two less transistors than gate C3, thus causing only the slightest difference in area and active power. Since gate C3 has a higher delay then gate C4, the PDP will also have a proportional overhead of 1.81 and 1.56 times greater than the PDP of the all custom CMOS. The mapping of gate C2 results in an active power and a PDP of 1.90 and 1.33 times greater than the all custom CMOS and an area overhead of 1.76 times larger.

As for the critical output, only three gates displayed a change in the critical path: gates C2, C4, and S2. Interestingly, gate C2 did not have a single critical output, but had multiple outputs which were S0, S2, and S3. Gates C4 and S2 are respectively a carry and sum output gate so their critical path changed to their own gates. The critical path changed for gate C4 due to the large LUT implementation and cause in delay. As for gate S2, a possibility as to why the critical path changed was due to its C2 input. Gate C2 is a 3-input gate and one of its inputs is an output from the critical path in which the delay may have carried over.

## Multiple Gate Implementation Results

For multiple gate mapping, certain optimization algorithms are used to select a set of gates to be replaced with LUTs. The optimization algorithms are:

Equations 1, 2, and 3 represent the independent selection, dependent selection, and parametric-aware dependent selection, respectively. The gates are chosen by following the specific rules of the algorithms.

Equation 1 represents the independent selection. *N* is the number of required test clocks (the amount of time) to determine all LUT implemented gates, *i*, (also called missing gates) chosen. *M* is the number of missing gates. *α* is the number of required patterns to determine a missing gate. *D* is the depth of the circuit which is the maximum number of flip-flops (also called latches, which are stable states of a circuit, usually two, that can withold memory) in a certain timing path. Equation 2 represents the dependent selection. *N*, *i*, *M*, *α*, and *D* are the same definition as in equation 1. *P* is the amount of possible gates for the missing gate. Equation 3 represents the parametric-aware dependent selection. The same parameters apply as in equations 1 and 2. *I* is the possible input(s) that work on the missing gates. (**Winograd**)

The independent selection are gates that do not intersect with each other nor are subsequent to each other. The algorithm applied to this selection chose specific gates with the 4-bit data inputs (A0 to A3, B0 to B3) and/or a sum output gate (S0 or S1). For inputs (A0, B0), gate AXB0 or the sum output gate S0 can be chosen. For inputs (A1, B1), gate AXB1 or the sum output gate S1 can be chosen. For inputs (A2, B2), only gate AXB2 can be chosen. For inputs (A3, B3), gates GB3, PB3, or AXB3 can be chosen. There are 12 possible combinations. Two examples of combinations are shown in figure 11.

|  |  |
| --- | --- |
| 1. One possible combination: gates GB3, AXB2, AXB1, and S0. | (b) Another possible combination: gates PB3, AXB2, AXB0, and S1. |

Figure 11. Two possible sets of independent seleted gates.

The dependent selection are gates that along the same path and can be subsequent to each other. The algorithm applied to this selection chose specific gates with delay overhead. Two examples of the selection are shown in figure 12. Figure 12(a) refers to the first case selection of gates GB0, N7, C2, and S2. This selection shows the dependency of gates when all have delay overhead and when the path is nonconsecutive. Gate N7 is dependent on gates GB0 and C0B. Gate C2 is dependent on one of three inputs, none of which were implemented with an LUT, and the output gate S2 is dependent on gate C2. In other words, this nonconsecutive path contains a break in dependency. Figure 12(b) refers to the second case selection of gates GB1, N5, N3, N9 and C4. Both the input and output gates, GB1 and C4, have delay overhead. This selection shows the dependency of gates when there is a mixture of gates with and without delay overhead and when the path is consecutive.

|  |  |
| --- | --- |
| 1. First case selection: gates GB0, N7, C2, and S2. | (b) Second case selection: gates GB1, N5, N3, N9 and C4. |

Figure 12. Two sets of dependently selected gates.

The parametric-aware dependent selection are gates that do not severely impact the delay, regardless of dependency. The algorithm applied to this selection chose all gates that do not produce delay overhead except for two 2-input gates, GB1 and GB0, and the carry output gate C4. The example of the selection is shown in figure 13.

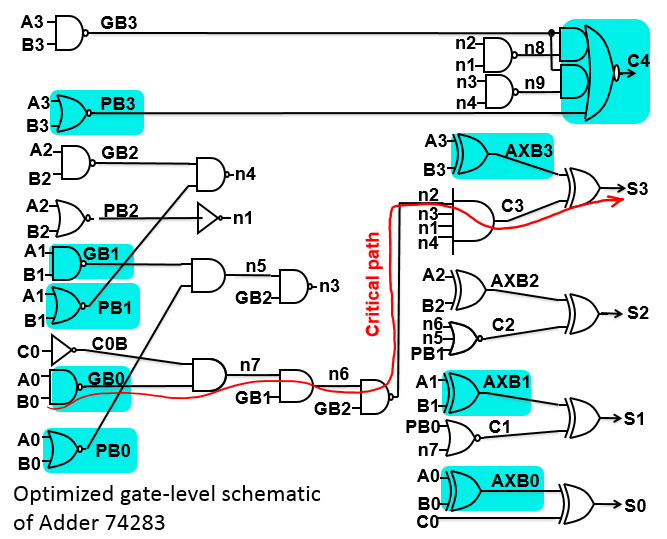


Figure 13. A set of parametric-aware dependent selected gates.

As can be concluded from Figure 14, LUT replacement of multiple gates does not result in delay overhead for independently selected gates. Based on the individual mapping results, the following gates chosen for this selection method did not produce delay overhead as well. LUT replacement of multiple gates result in delay overhead for both dependent and parametric-aware dependent selections. The first case of dependent selection resulted in a delay of 32 ps, which is 1.23 times greater than the all custom CMOS. The second case of dependent selection resulted in a delay of 30 ps, which is 1.15 times greater and the parametric-aware dependent selection resulted in a delay of 29 ps, which is 1.12 times greater. There is more overhead in the dependent selection, despite having lesser gates implemented than the parametric-aware dependent. When gates in the same timing path are replaced with LUTs, the input of one gate is dependent on the other, thus it is possible for more delay to carry over. The results are shown in a bar graph in figure 14.

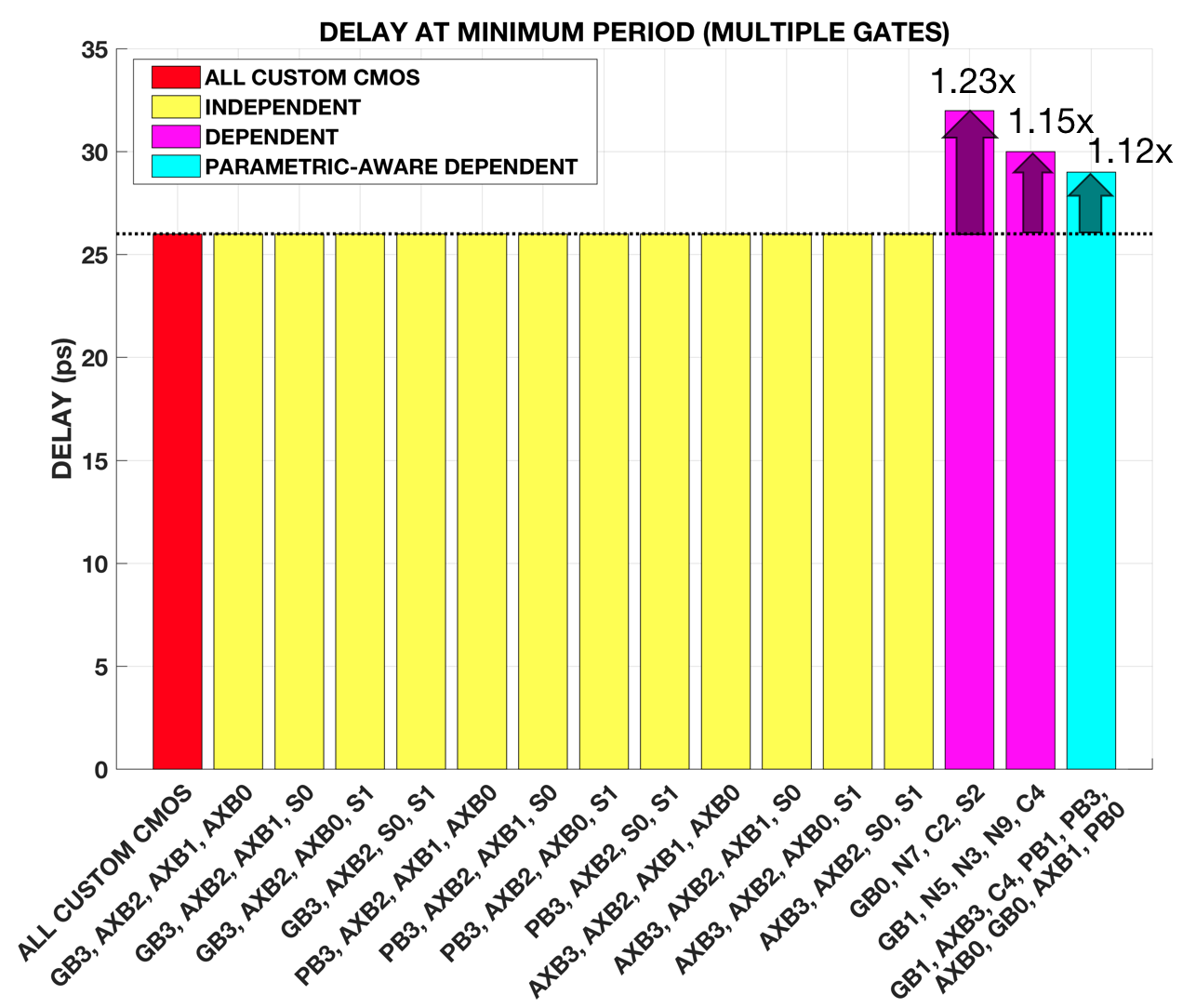


Figure 14. Delay at minimum period (multiple gate mapping).

LUT replacement of multiple gates produce in active power, area, and PDP overhead. The results are shown in a set of bar graphs at figures 15, 16, and 17, respectively. The results are extracted in a similar way with the individual mapping data. Upon comparing the results of the independent selections, it is noticeable that they have roughly the same values for in terms of active power, area, and PDP. All independent selections are approximately 2.68 times greater in active power than the all custom CMOS. All independent selections are approximately 2.34 and 1.40 times greater in area and PDP, respectively, than the all custom CMOS. The resason for this trend is because all the selected gates are 2-input gates and all are replaced with LUT2.

With increasing area, there will also be an increase in active power and PDP. The first and second dependent selection and the parametric-aware selection had an area overhead of 2.82, 4.03, and 5.35 times greater, respectively. The same trend is observed for the active power, which are 3.15, 4.48, and 6.21 times greater, and the PDP, which are 1.68, 2.01, and 2.46 times greater than the all custom CMOS.

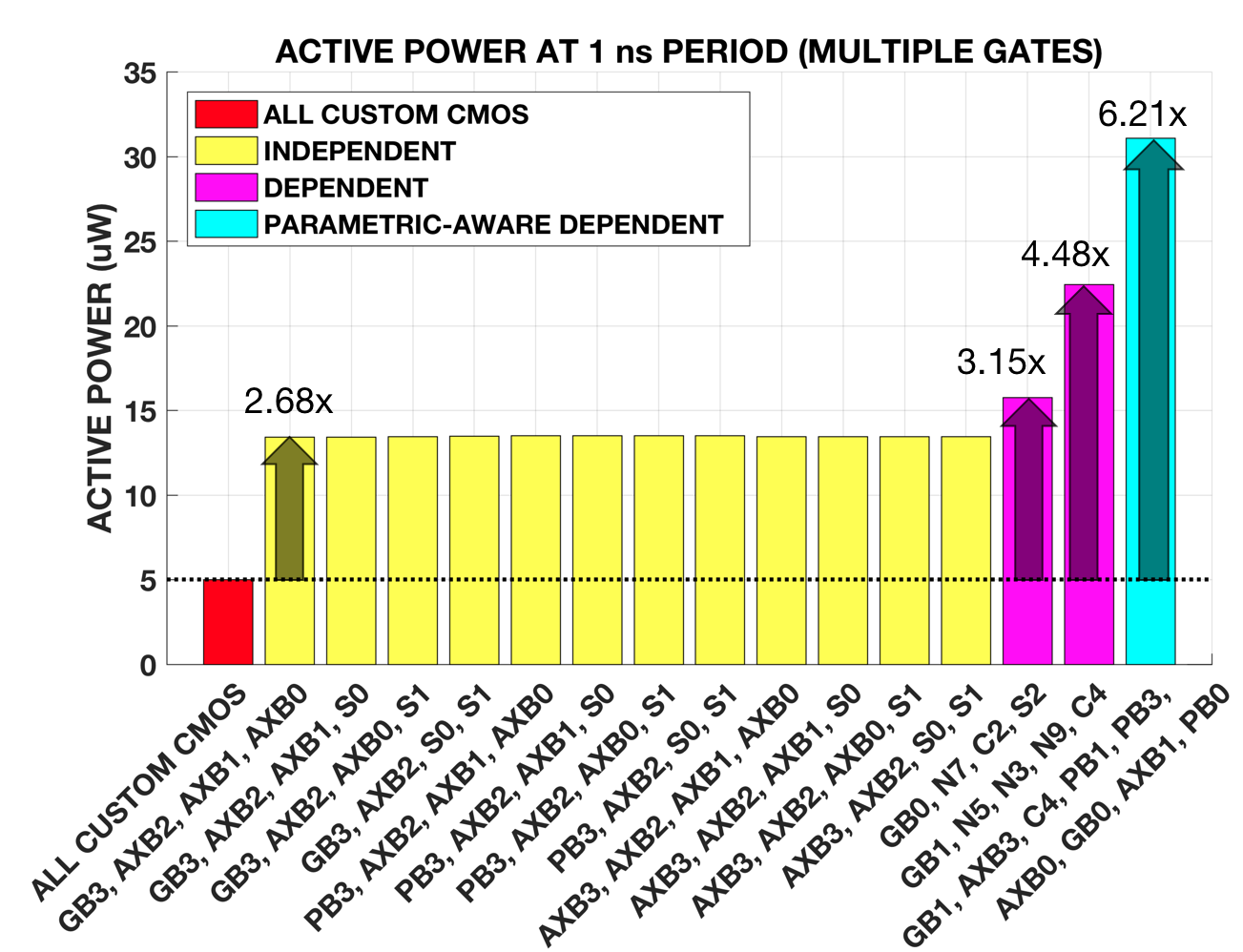


Figure 15. Active power at a 1 ns period (multiple gate mapping).

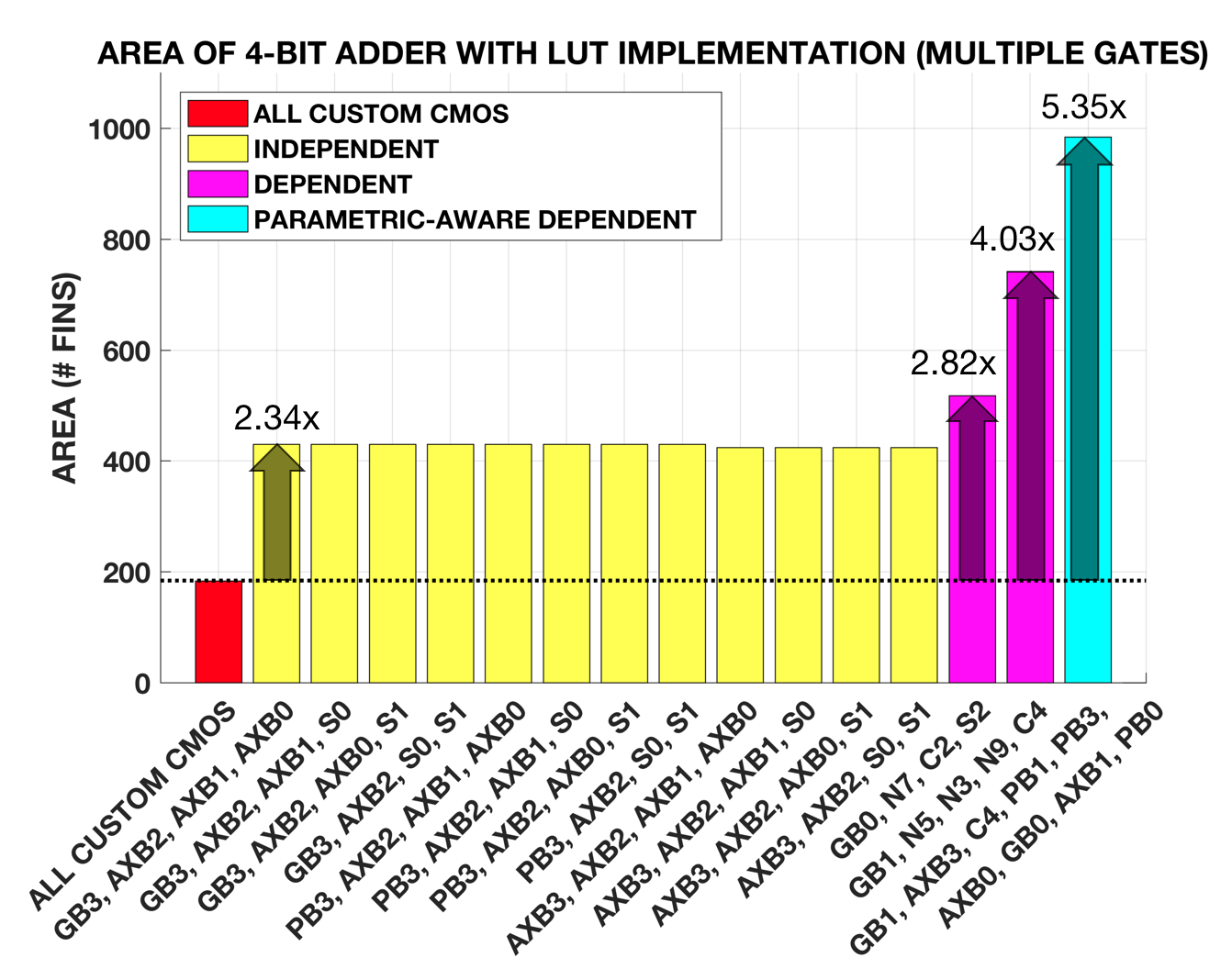


Figure 16. Area of 4-bit adder (multiple gate mapping).

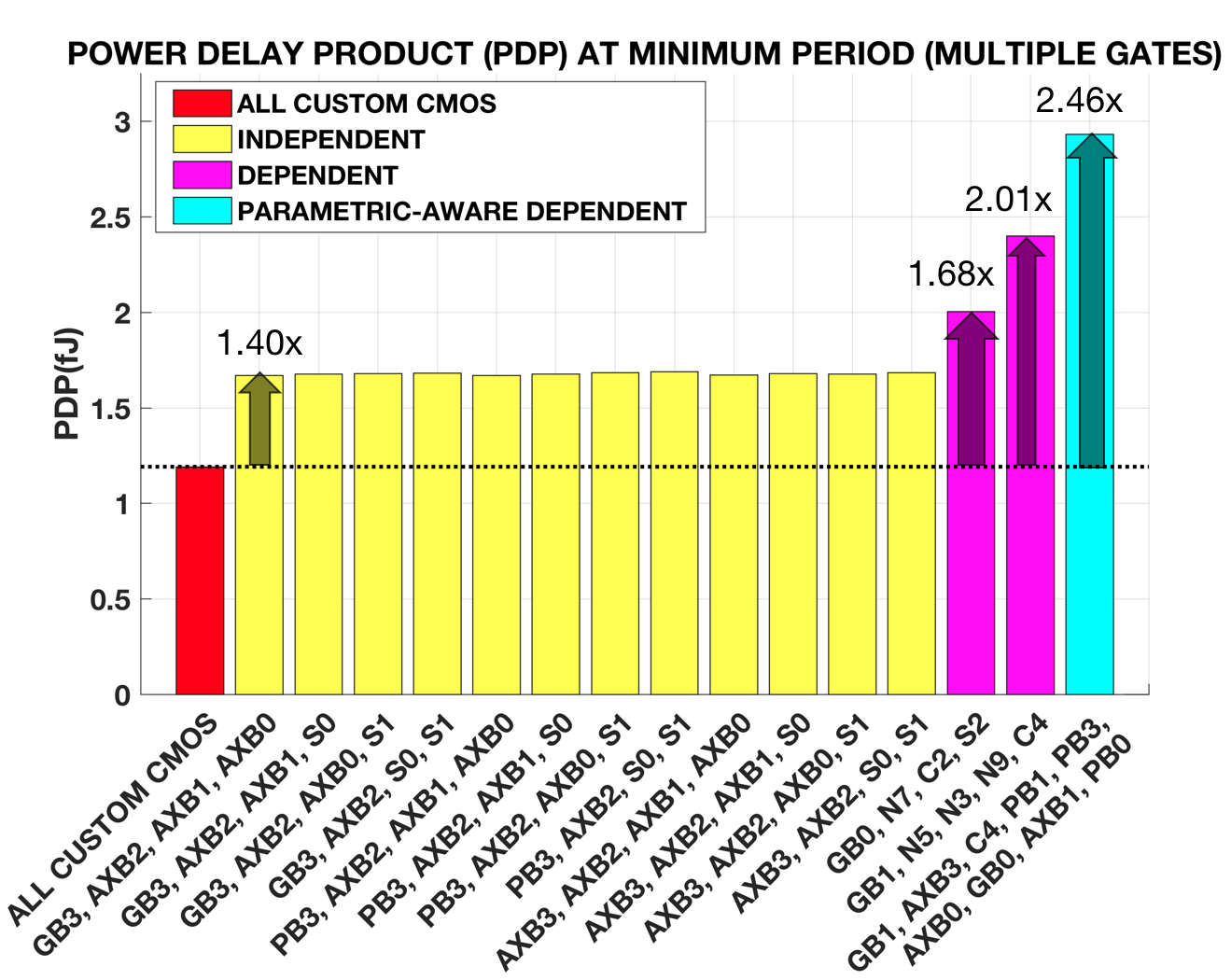


Figure 17. PDP at minimum period (mulitple gate mapping).

The critical output was also observed for the multiple gate mapping. The independent selections did not have a change in critical path, as their critical output still ends at gate S3. For the first dependent selection, the critical output was gate S2. A possibility for this change is due to the first two gates, GB0 and N7, being in the critical path, and gate C2, which is a 3-input gate, thus providing more delay. For the second dependent and the paramteric-aware dependent selections, their critical output was gate C4. Since gate C4 was implemented with an LUT and there was an observed high delay from the individual results, the critical path changed to this output.

**Conclusion**

As expected, gates in the

The independent selection avoided delay overhead and although it is more favorable for the circuit’s performance, it is however more vulnerable to an attacker. The dependent selection produced more delay overhead than the parametric-aware dependent selection, this is due to LUT implementation on gates along the same timing path.

The parametric-aware dependent selection requires more gates for LUT implementation, however it has a lower delay compared to the dependent selections because it chooses gates that do not violate the timing path.