Problem 1

Perform the following six tasks for
a. 8-bit Hybrid Brent-Kung/Kogge-Stone Parallel Prefix Network Adder,
b. 8-bit Conditional Sum Adder, and
c. 8-bit Ripple Carry Adder.
Each adder should be capable of performing operation
\[(c_8, S) = X + Y + c_0,\]
where X, Y, and S are 8-bit unsigned integers, and c_0 represents one-bit carry in.

Task 1

Draw full block diagrams of all adders. You can draw all adders by hand or electronically. The electronic version will be rewarded with bonus points.

For the Ripple Carry Adder use Full Adders as basic building blocks. For remaining two adders use basic building blocks suitable for each adder.

Show the internal structure of each basic building block using inverters and arbitrary logic gates with up to four inputs.

Task 2

Using block diagrams developed in Task 1, show all intermediate values in each node of each adder when performing the operation \( X + Y + c_0 \) for the following values of inputs:
\[
\begin{align*}
X &= 10101011 \\
Y &= 10010111 \\
c_0 &= 1
\end{align*}
\]
Verify that the obtained result, including cout = c_8, is correct and the same for all adders.

Task 3

Using block diagrams developed in Task 1, show the moment in time, t, when the computed value appears in each node of your circuit, assuming the worst case values of inputs X, Y, and c_0. Assume that all inputs to the adders are available at time t=0. Assume that the delay of each gate and the delay of an inverter is equal to 1.

Task 4

Calculate the Latency and Area of each adder, under the following assumptions:
- all inputs, including control inputs, are registered (outside of your circuit)
• delays of all gates are independent of the number of inputs, and are equal to the delay of an inverter. Thus, you can assume that the delay of each gate and inverter is equal to 1.
• area of each gate is proportional to the number of inputs. For example, area of a 2-input gate is equal to the area of 2 inverters. Assume that the area of an inverter is equal to 1, area of a 2-input gate is equal to 2, etc.
• for a D-flip-flop, the clock-to-output delay is equal to 2, and its setup time is equal to 1; please include these timing parameters in the calculation of your Latencies.
• do not include the areas of the surrounding input and output registers in your computations.

Task 5

Develop VHDL or Verilog code for a generic k-bit version of each aforementioned adder. For the Ripple Carry Adder use “+” for its implementation. Do not build this adder out of Full Adders.

Develop generic k-bit wrappers that surround the combinational circuits listed above and store each input and output in a register.

Task 6

Synthesize and implement all three aforementioned circuits for the following values of the generic k:

\[ k = 2^i \quad \text{for} \quad i=3, 4, 6, 8. \]

targeting Xilinx Spartan 3 FPGAs. Use the smallest device of the Spartan 3 family capable of holding each adder for \( k=2^8=256 \).

When determining circuit area (measured in CLB slices), synthesize and implement your circuit without any surrounding registers.

When determining minimum clock periods, synthesize and implement your circuit with surrounding input/output registers. Assume that the latency of each investigated circuit is equal to the minimum clock period obtained from your implementation.

Task 7

Create tables and draw diagrams showing latencies, areas (in CLB slices), and latency*area products of all three aforementioned adders as a function of k, with \( k = 2^i \), for \( i=3, 4, 6, 8 \).

Analyze your results, and describe your observations and conclusions in the final report.