ECE385 Experiment #4

Eric Meyers, Ryan Helsdingen

Section ABG; TAs: Ben Delay, Shuo Liu

February 17th, 2016

emeyer7, helsdin2

I. Introduction

THE purpose of this lab was to get a brief introduction to System Verilog and Quartus by creating and analyzing a total of 4 designs. The first is a 8 bit serial logic processor that was created during last lab (Lab 3) using TTL chips on a breadboard. The next three being different configurations of 16-bit adders: A carry-ripple adder, a carry-lookahead adder, and a carry-select adder. The team then analyzed the performance of these three adders by selecting different metrics to compare as well as analyzed the performance of the serial logic processor compared to the design used in Lab 3.

II. Schematic of Logic Processor

Please refer to Figure 2 in ”Section XI: Figures” to view the Schematic of the 8-bit Logic Processor designed in the Pre-Lab of this Experiment.

III. Design Simulations of Logic Processor

Please refer to Figure 3 in ”Section XI: Figures” to view the Annotated RTL Simulation Output of the 8-bit Logic Processor designed in the Pre-Lab of this Experiment.

IV. Written Description of Adder Circuit

RYAN SECTION

V. Purpose of Modules

RYAN SECTION

VI. State Machine

The State Machine for the Serial Bit Logic Processor Control Unit is shown in Figure 5 in ”Section XI: Figures” in this document.

The State Machine is broken down into a total of ten states labeled A through J, with A being the initial/halt state and J being the final state. PUT MORE HERE

VII. Schematic Block Diagrams

RYAN SECTION

-I ADDED BLOCK SCHEMATICS FROM QUARTUS IN FIGURES SECTION - you just need to put descriptions here of what the figures mean down below / add your high level block diagrams if youd like

VIII. Design Analysis Comparison

The following table displays the values the team received upon analyzing the metrics requested in the pre-lab.

Metric	Ripple	Lookahead	Select
Memory(BRAM)	0	0	0
Frequency (MHz)	62.81	64.526	61.55
Power (mW)	156.65	156.39	156.30

The following figure

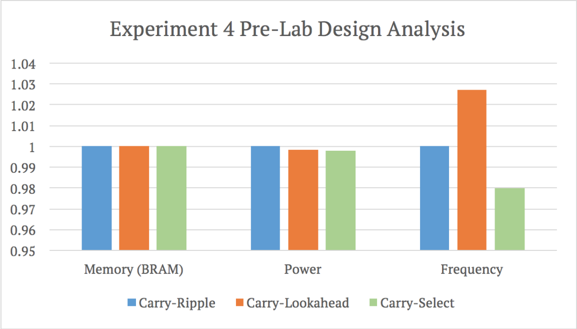
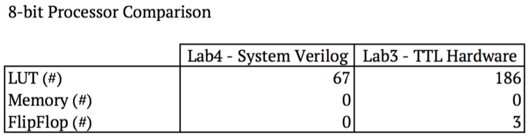


Fig. 1: Area, Power, Frequency Comparison

IX. Post-Lab

1) Compare the usage of LUT, Memory, and Flip-Flop of your bit-serial logic processor exercise in the IQT with your TTL design in Lab 3. Make an educated guess of the usage of these resources for TTL assuming the processor is extended to 8-bit. Which design is better, and why?

Answer: The System Verilog design used in Lab 4 clearly dominates the total usage of Logic Elements in Lab 3 almost by a factor of three. This is because the system used in Lab4 is very efficient and optimizes/reduces the amount of Logic Elements used as neccessary. However, this was not done in Lab 3 efficiently. If the team attempted to do Lab 3 again, a lower LE count would have been achieved but no where near the efficiency of using Quartus.



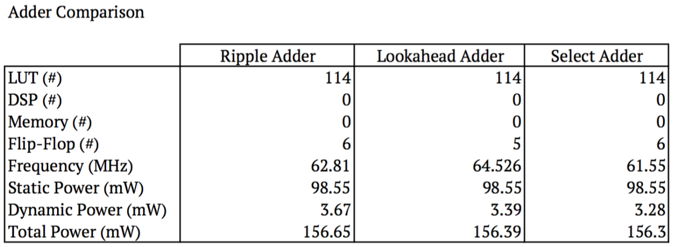
8-bit Processor Comparison

	Lab4 - System Verilog	Lab3 - TTL Hardware
LUT (#)	67	186
Memory (#)	0	0
FlipFlop (#)	0	3

TABLE I: Processor Area Comparison

2) For the adders, refer to the Design Resources and Statistics in IQT.30-32 and complete the following design statistics table for each adder. This is more comprehensive than the above design analysis and is required for every SystemVerilog circuit.

These numbers are shown below in Table II .



Adder Comparison

	Ripple Adder	Lookahead Adder	Select Adder
LUT (#)	114	114	114
DSP (#)	0	0	0
Memory (#)	0	0	0
Flip-Flop (#)	6	5	6
Frequency (MHz)	62.81	64.526	61.55
Static Power (mW)	98.55	98.55	98.55
Dynamic Power (mW)	3.67	3.39	3.28
Total Power (mW)	156.65	156.39	156.3

TABLE II: Processor Area Comparison

Observe the data plot and provide explanation to the data, i.e., does each resource breakdown comparison from the plot makes sense? Are they complying with the theoretical design expectations, e.g., the maximum operating frequency of the carry-lookahead adder is higher than the carry-ripple adder? Which design consumes more power than the other as you expected, why?

Answer: The frequency of the Lookahead Adder was, indeed, faster than that of the Ripple Adder or the Select Adder. This makes sense because, although the Lookahead Adder consumes more area through logic elements (must connect each adder together with some logic so that the next adder can see the carry bits), it is faster which is shown in the frequency analysis.

The only result that was somewhat surprising to the group was the fact that the Select Adder was slower than the Ripple Adder. The team was not sure why this was the case because the Ripple Adder should *usually* be slower than the Ripple Adder however, not in this case. This might be attributed to the fact that.....

X. Conclusion

Overall, the team successfully demonstrated all three different adder configurations (Carry-Ripple, Carry-Lookahead, and Carry-Select) during the demo period for full credit. These adders were analyzed in the previous sections and it is evident that the Lookahead Adder was the fastest out of all three, and the Select Adder consumed the least amount of power out of all three.

The serial bit logic processor created in the pre-lab outperformed the bit logic processor constructed in lab 3 in several manners. For one, the total number of logic elements on the System Verilog Processor was $\approx$ 60 while the TTL Lab 3 total logic elements was exceeding $\approx$ 180. This is almost a factor of three reduction because of Quartus optimization techniques and improvements to the circuit.

Overall, the team gained plenty of insight to the benefits of using System Verilog vs. using TTL chips on breadboards. The complexity decreased significantly and breaking sequential and combinational logic into modules through System Verilog proves to be a very effective.

XI. Figures

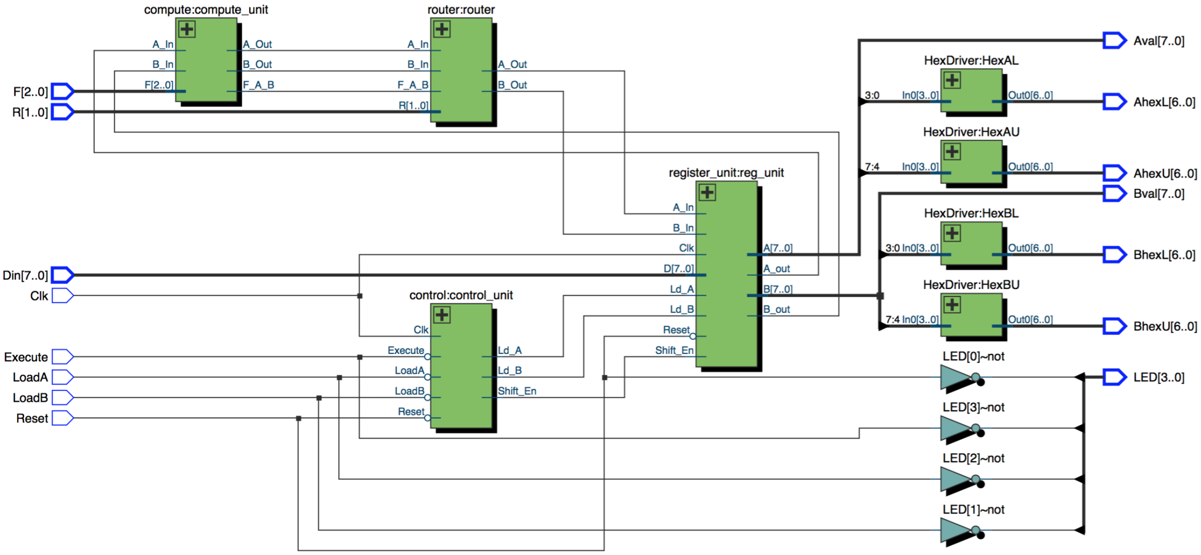
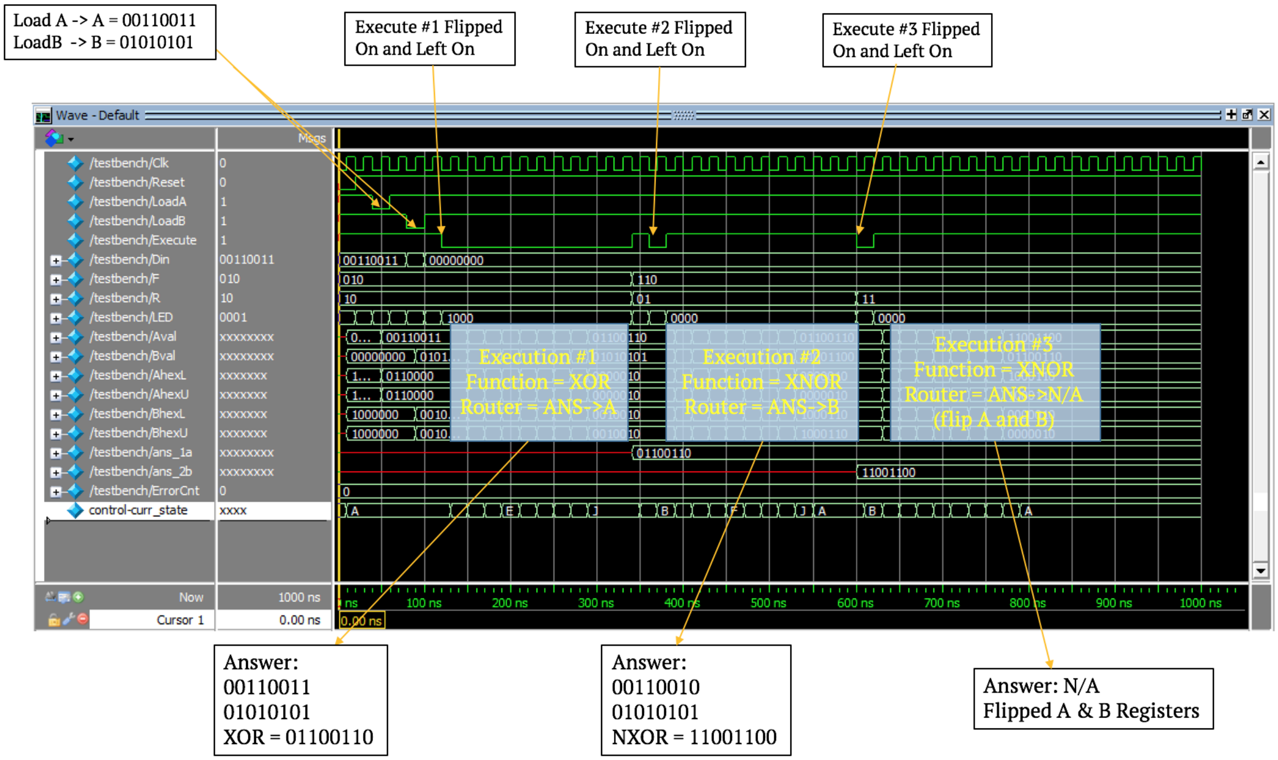


Fig. 2: Serial Logic Processor Schematic



Load A -> A = 00110011
LoadB -> B = 01010101

Execute #1 Flipped
On and Left On

Execute #2 Flipped
On and Left On

Execute #3 Flipped
On and Left On

Execution #1
Function = XOR
Router = ANS->A

Execution #2
Function = XNOR
Router = ANS->B

Execution #3
Function = XNOR
Router = ANS->N/A
(flip A and B)

Answer:
00110011
01010101
XOR = 01100110

Answer:
00110010
01010101
NXOR = 11001100

Answer: N/A
Flipped A & B Registers

Fig. 3: Serial Logic Processor RTL Simulation Output

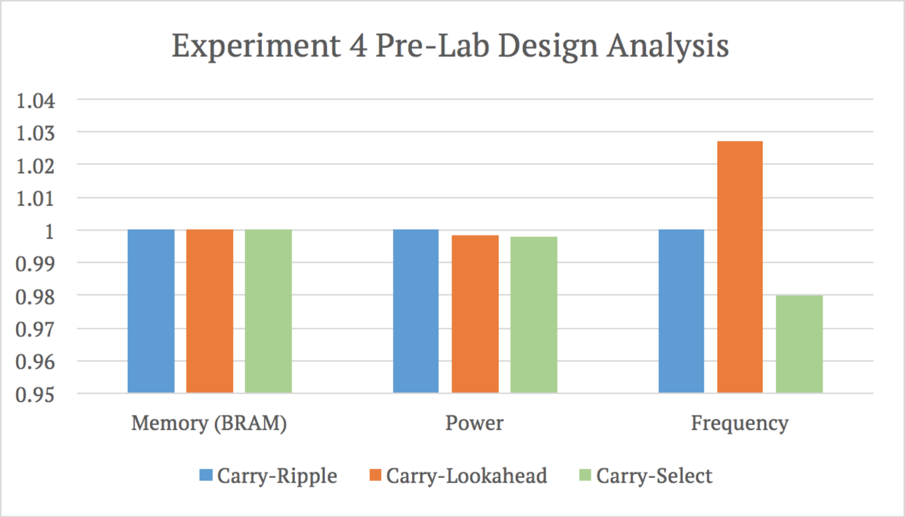


Fig. 4: Pre-Lab Adder Design Analysis (Area, Power, Frequency)

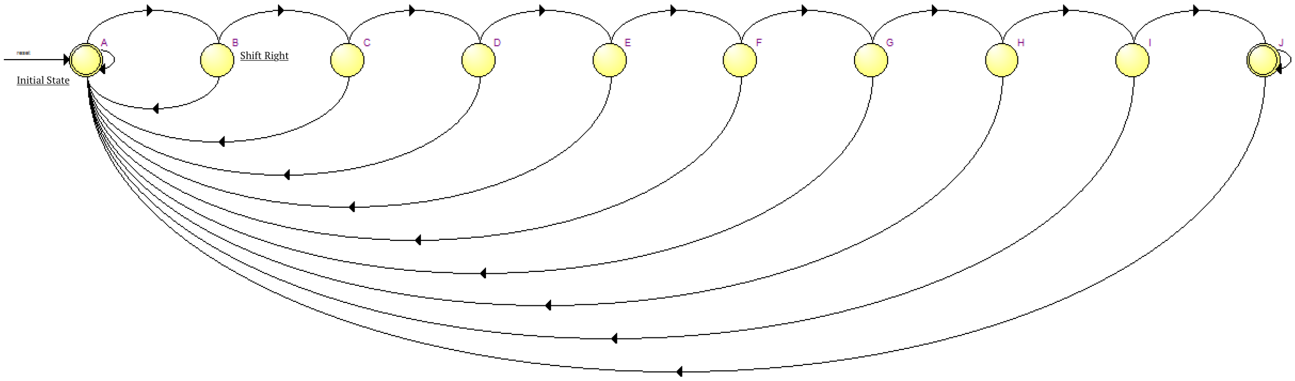


Fig. 5: State Machine for Serial Bit Logic Processor

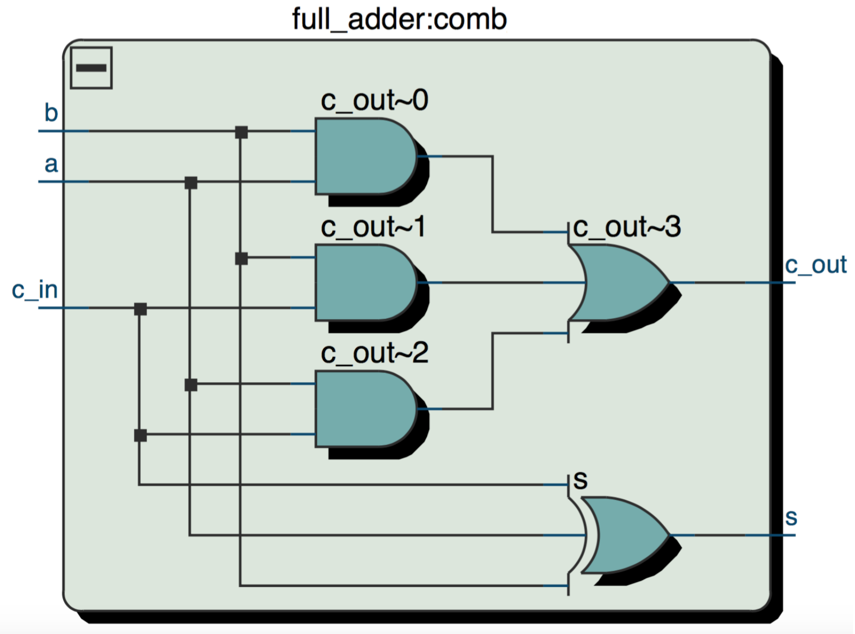


Fig. 6: Full-Adder Block Schematic

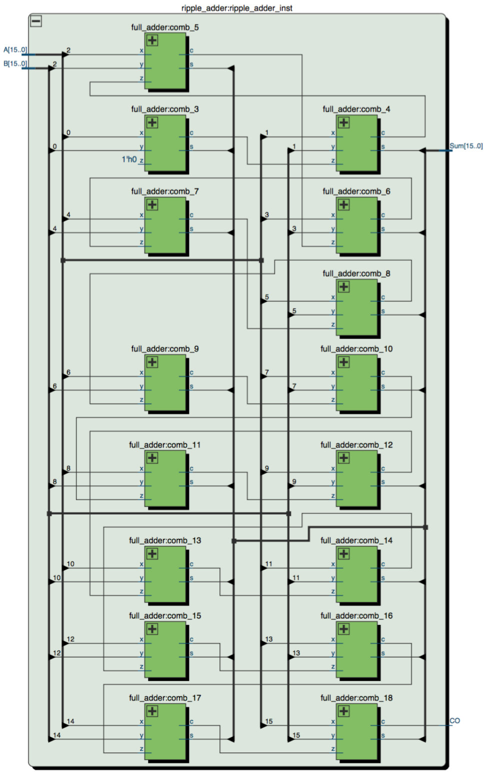


Fig. 7: Carry Ripple Block Schematic

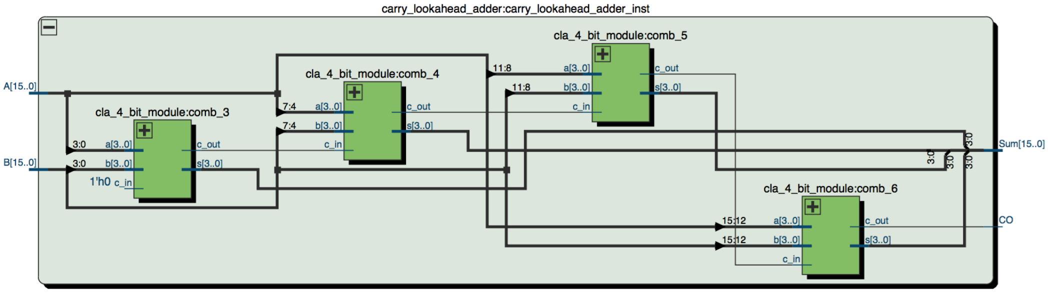


Fig. 8: Carry Lookahead (CLA) Block Schematic

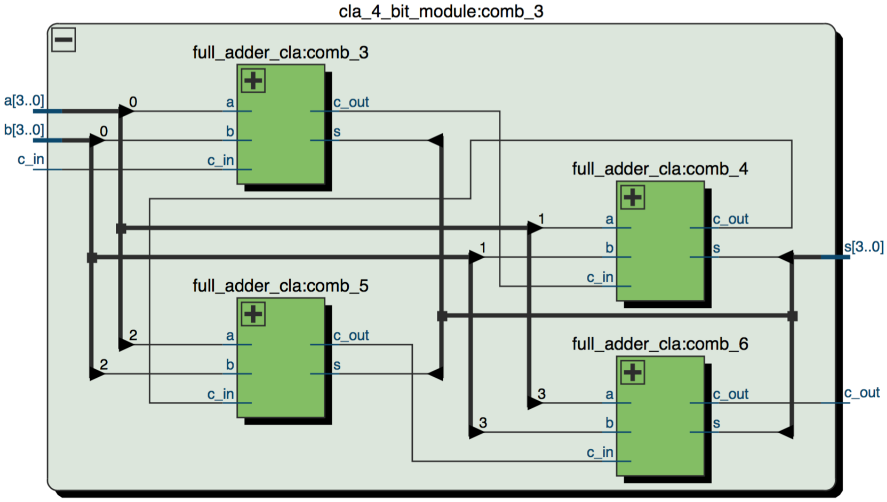


Fig. 9: Four-Bit CLA Block Schematic

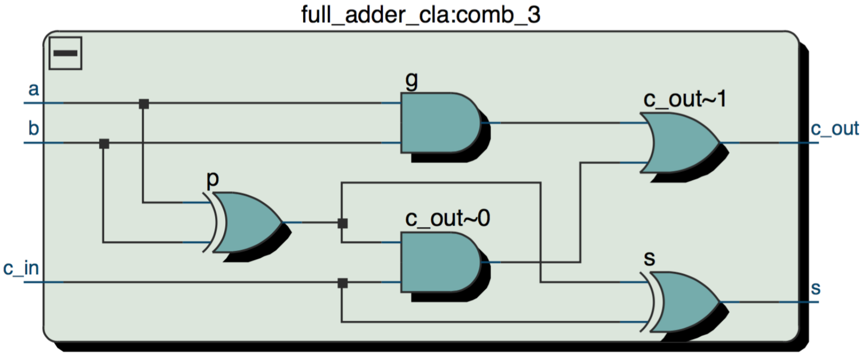


Fig. 10: One-Bit CLA Block Schematic

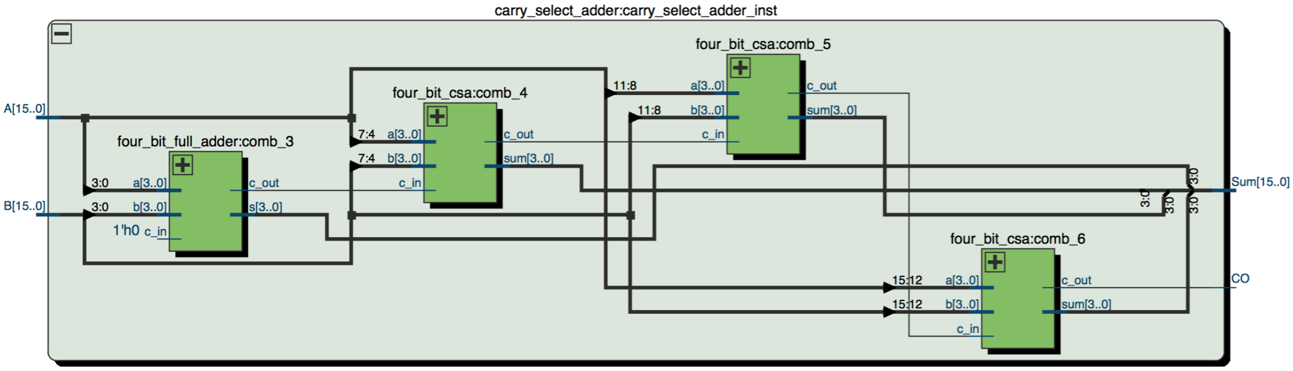


Fig. 11: Carry Select (CSA) Block Schematic

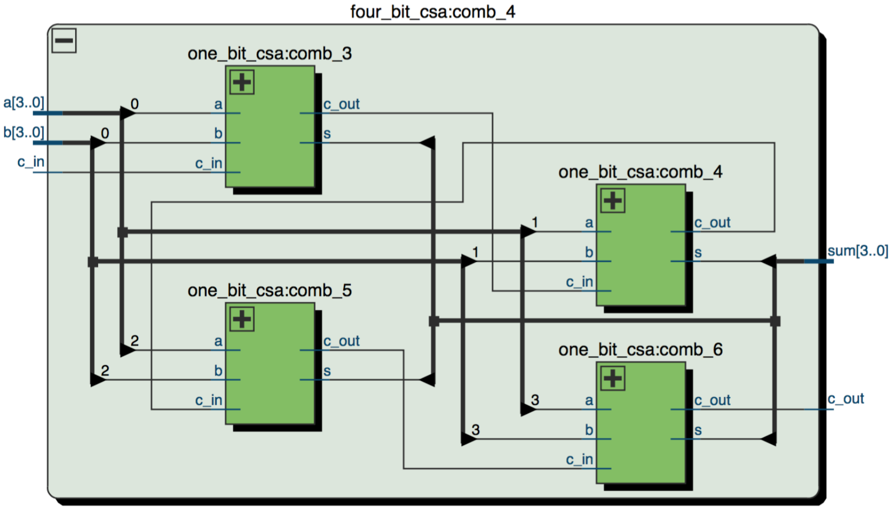


Fig. 12: Four-Bit CSA Block Schematic

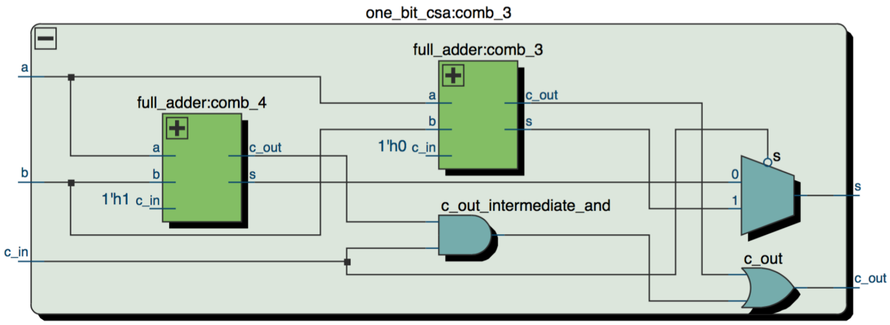


Fig. 13: One-Bit CSA Block Schematic