Laboratory Report # 2

Name: Lauron, John Enrico D. **Date Completed:** August 29, 2025

Laboratory Exercise Title: Basic Constructs in Verilog HDL

Target Course Outcomes:

CO1: Create descriptions of digital hardware components such as in combinational and sequential circuits using synthesizable Verilog HDL constructs.

CO2: Verify the functionality of HDL-based components through design verification tools.

Exercise 2B: Full Adder

In this exercise, we designed a 1-bit full adder using Verilog HDL with structural modeling and gate primitives. The circuit adds two inputs (x, y) and a carry-in (Cin) to produce the sum (S) and carry-out (Cout). Furthermore, the circuit was simulated using a testbench file to verify the generated waveform.

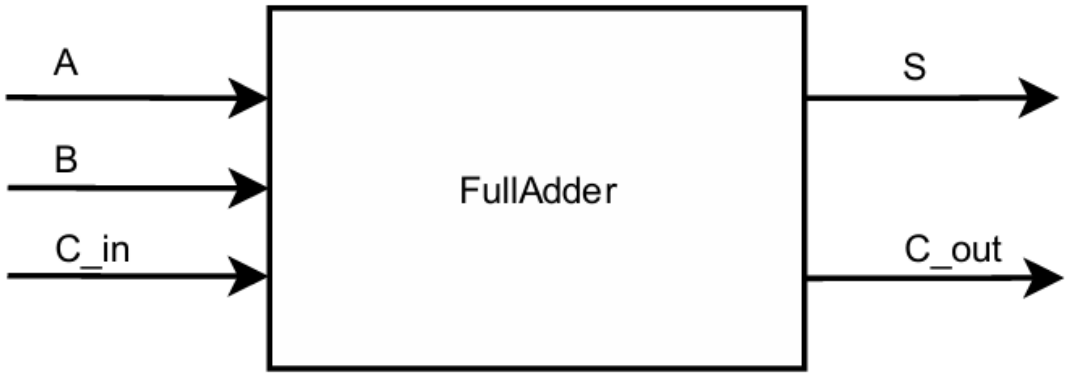


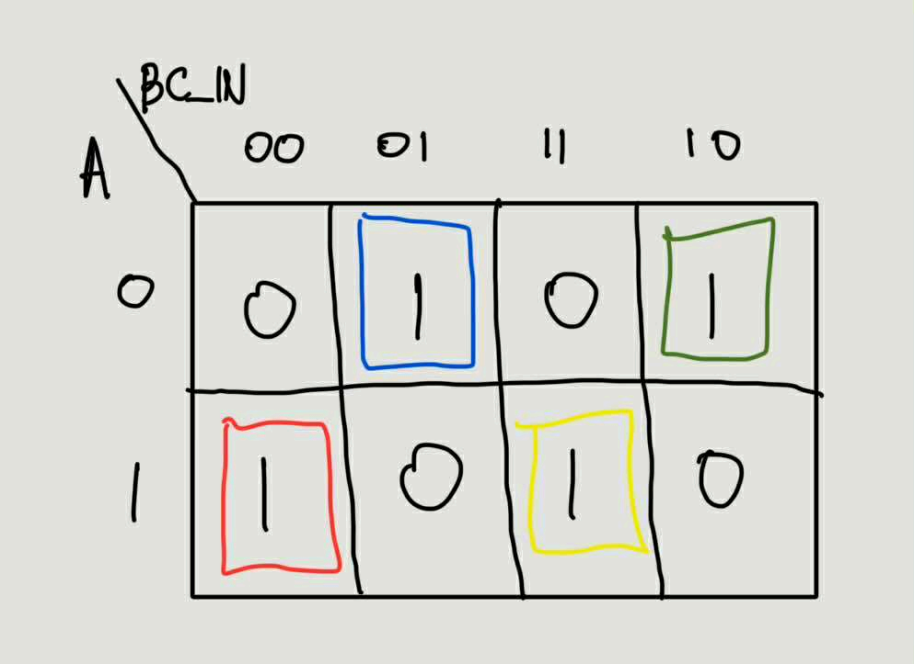
Figure 1. Entity Diagram of a Full Adder

I / O	Variable	Description
Inputs	A	*First single-bit input*
	B	*Second single-bit input*
	C_in	Carry-in
Outputs	C_out	*Carry-out*
	S	Sum

Table 1. Full Adder Circuit I/O

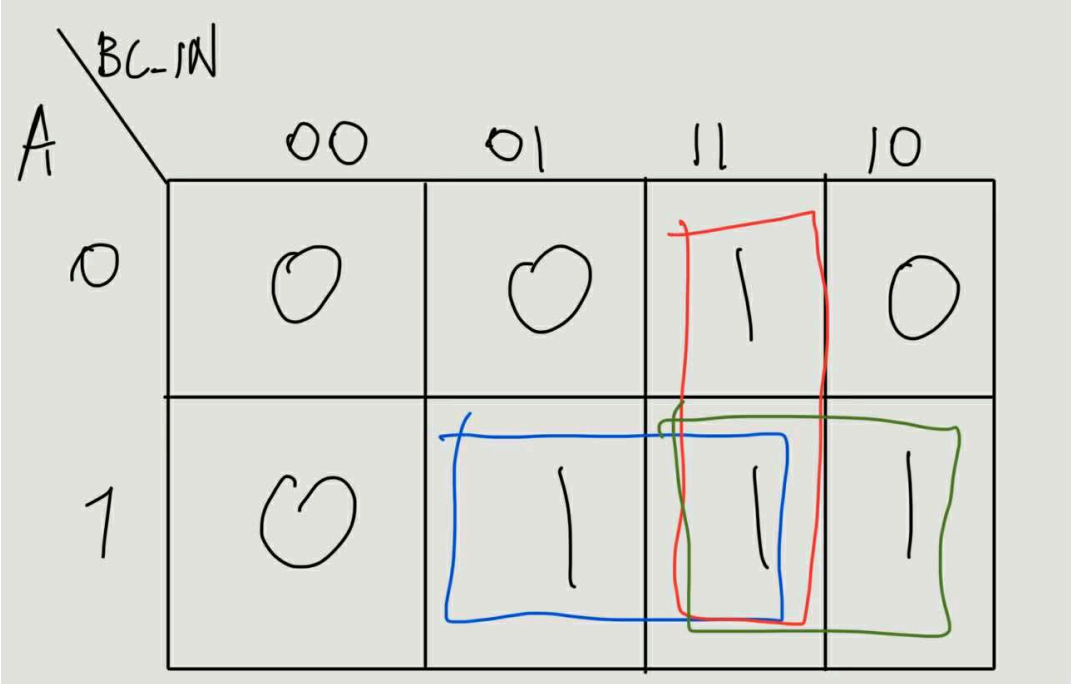
A	B	C_IN	S	C_OUT
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Table 2. Full Adder Circuit Truth Table



$$S = A \oplus B \oplus C\_IN$$

Figure 2. K-Map of S Output



C_out = AB + C_IN (A ⊕ B)

Figure 3. K-Map of C_Out Output

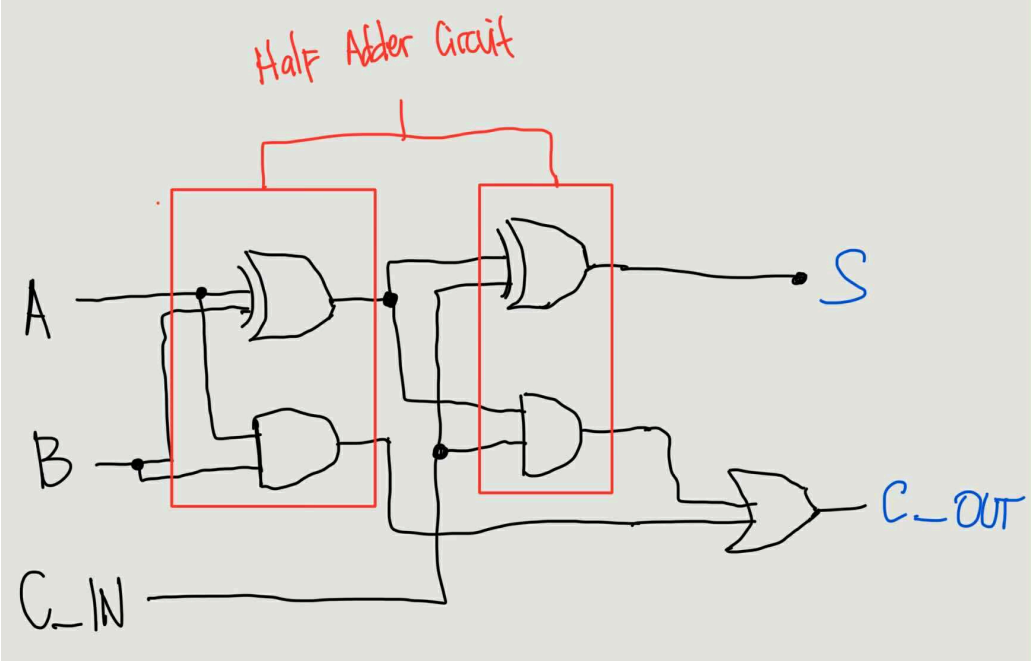


Figure 4. Full Adder Circuit Logic Diagram (with annotations)

Flow Summary

<<Filter>>

Flow Status	Successful - Sat Aug 30 00:13:28 2025
Quartus Prime Version	20.1.1 Build 720 11/11/2020 SJ Lite Edition
Revision Name	FullAdder
Top-level Entity Name	FullAdder
Family	MAX 10
Device	10M50DCF484C7G
Timing Models	Final
Total logic elements	2
Total registers	0
Total pins	5
Total virtual pins	0
Total memory bits	0
Embedded Multiplier 9-bit elements	0
Total PLLs	0
UFM blocks	0
ADC blocks	0

Figure 5. Flow Summary of Full Adder

Type	ID	Message
⚠	18236	Number of processors has not been specified which may cause overloading on shared machines. Set the global assignment NUM_PARALLEL_PROCESSORS in your QSF to an appropriate va
ℹ	20030	Parallel compilation is enabled and will use 4 of the 4 processors detected
ℹ	12021	Found 1 design units, including 1 entities, in source file tb_halfadder.v
ℹ	12021	Found 1 design units, including 1 entities, in source file halfadder.v
ℹ	12021	Found 1 design units, including 1 entities, in source file fulladder.v
ℹ	12127	Elaborating entity "FullAdder" for the top level hierarchy
ℹ	12128	Elaborating entity "HalfAdder" for hierarchy "HalfAdder:HA1"
ℹ	286030	Timing-Driven Synthesis is running
ℹ	16010	Generating hard_block partition "hard_block:auto_generated_inst"
ℹ	21057	Implemented 7 device resources after synthesis - the final resource count might be different
ℹ		Quartus Prime Analysis & Synthesis was successful. 0 errors, 1 warning

Figure 6. Synthesis Result of Full Adder

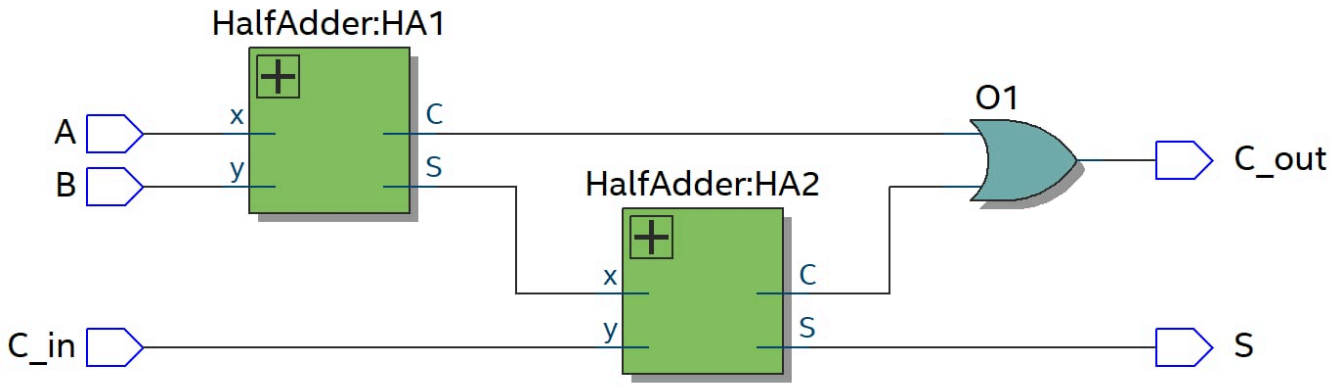


Figure 7. RTL View of Full Adder

FullAdder.v

```
1  /*********************************
2  * File:           FullAdder.v
3  * Author:         John Enrico Desalago Lauron
4  * Class:          CPE 3101L
5  * Group/Schedule: Group 1 Fri 7:30 - 10:30 AM
6  * Description:    Verilog HDL Code of FullAdder
7  *********************************/
8
9  module FullAdder (A, B, C_in, S, C_out);
10
11       input    A, B, C_in;
12       output   C_out, S;
13       wire     W1S, W2C, W3O;
14
15       HalfAdder HA1 (A, B, W2C, W1S);
16       HalfAdder HA2 (W1S, C_in, W3O, S);
17       or        O1   (C_out, W3O, W2C);
18
19 endmodule
20
```

Figure 8. Verilog HDL Code of Full Adder

FullAdder.v | Compilation Report - FullAdder | tb_FullAdder.v

```
1  /*******************************
2  * File:           tb_FullAdder.v
3  * Author:         John Enrico Desalago Lauron
4  * Class:          CPE 3101L
5  * Group/Schedule: Group 1 Fri 7:30 - 10:30 AM
6  * Description:    Testbench file for FullAdder.v
7  *******************************/
8
9  `timescale 1 ns / 1 ps
10 module tb_FullAdder ();
11
12    reg      A, B, C_in;
13    wire     C_out, S;
14
15    FullAdder UUT (A, B, C_in, S, C_out);
16
17    initial
18    begin
19       A = 0; B = 0; C_in = 0;    #10
20       A = 0; B = 0; C_in = 1;    #10
21       A = 0; B = 1; C_in = 0;    #10
22       A = 0; B = 1; C_in = 1;    #10
23       A = 1; B = 0; C_in = 0;    #10
24       A = 1; B = 0; C_in = 1;    #10
25       A = 1; B = 1; C_in = 0;    #10
26       A = 1; B = 1; C_in = 1;    #30
27
28       $stop;
29    end
30
31 endmodule
32
```

Figure 9. Verilog HDL Code of Testbench Full Adder

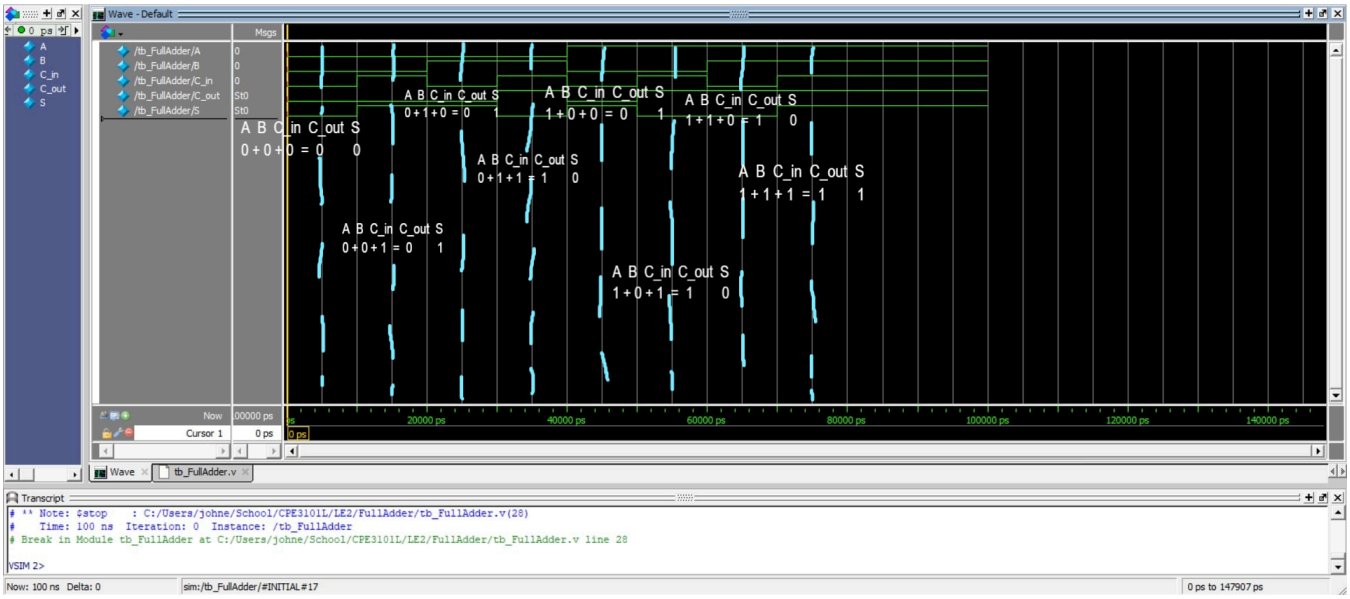


Figure 10. Testbench Waveform for Full Adder (with annotations)