EXP4:FULL ADDER AND FULL SUBTRACTOR

FULL ADDER AND SUBTRACTOR

AIM:

To design a Full Adder and Full Subtractor circuit and verify its truth table in Quartus using Verilog programming.

Equipments Required:

Hardware – PCs, Cyclone II , USB flasher

Software – Quartus prime

Theory:

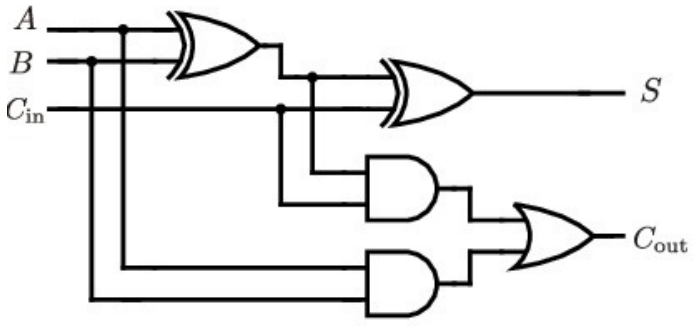
Full Adder:

Full adder is a digital circuit used to calculate the sum of three binary bits. It consists of three inputs and two outputs. Two of the input variables, denoted by A and B, represent the two significant bits to be added. The third input, Cin, represents the carry from the previous lower significant position. Two outputs are necessary because the arithmetic sum of three binary digits ranges in value from 0 to 3, and binary 2 or 3 needs two digits. The two outputs are sum and carry.

Sum =A'B'Cin + A'BCin' + ABCin + AB'Cin' = A ⊕ B ⊕ Cin

Carry = AB + ACin + BCin

Logic diagram:



Truth table:

Inputs			Outputs	
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

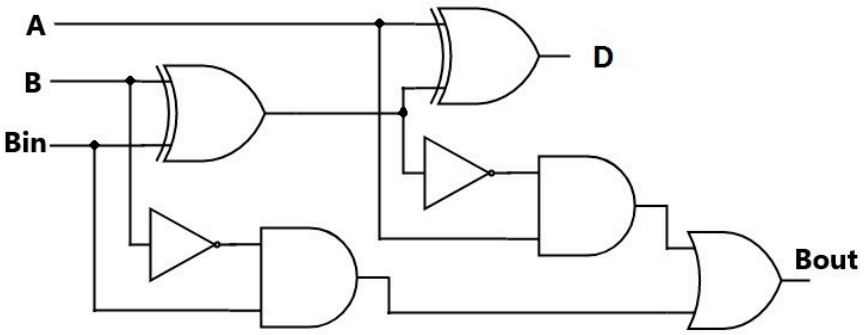
Full Subtractor

A full subtractor is a combinational circuit that performs subtraction involving three bits, namely minuend, subtrahend, and borrow-in . It accepts three inputs: minuend, subtrahend and a borrow bit and it produces two outputs: difference and borrow.

Diff = A ⊕ B ⊕ Bin

Borrow out = A'Bin + A'B + BBin

Logic diagram:



Truth table:

Inputs			Outputs	
A	B	$Borrow_{in}$	Diff	Borrow
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

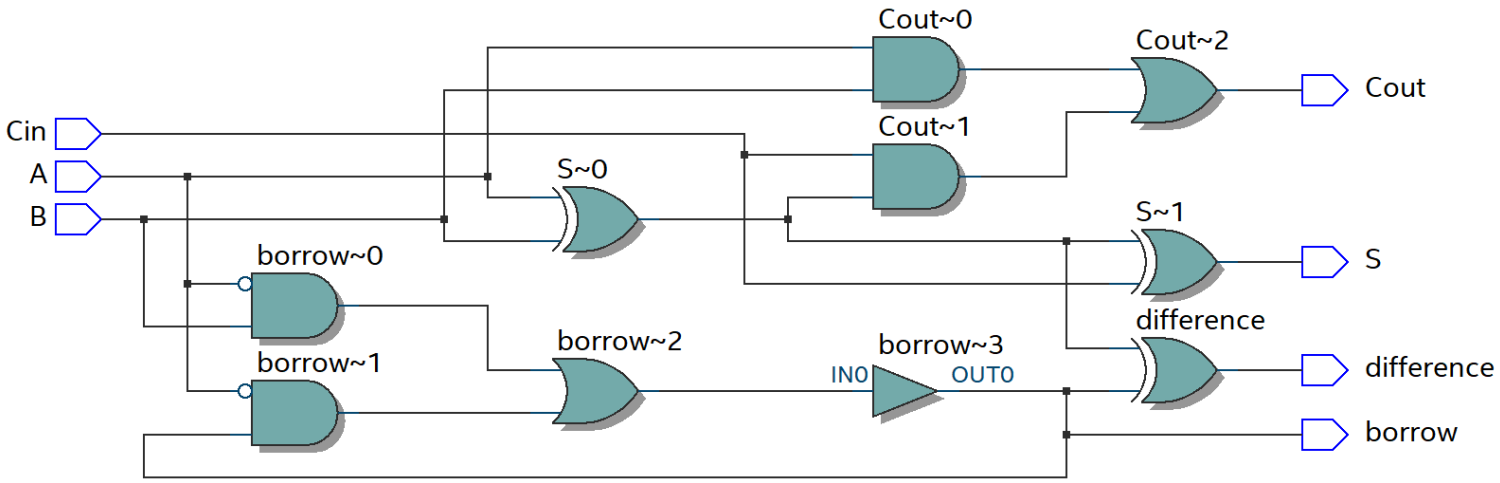
Procedure:

1. Type the program in Quartus software.
2. Compile and run the program.
3. Generate the RTL schematic and save the logic diagram.
4. Create nodes for inputs and outputs to generate the timing diagram.
5. For different input combinations generate the timing diagram.

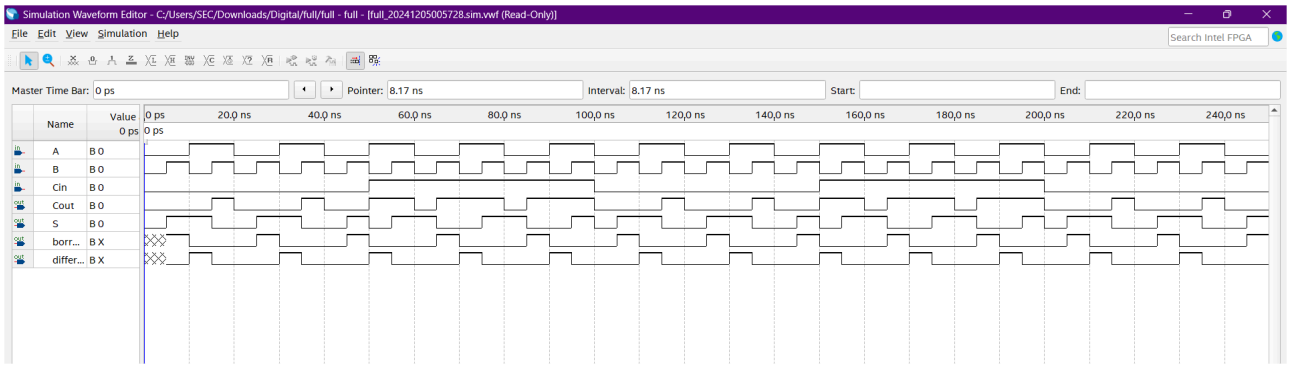
Program:

```
1  module full(A, B, Cin, S, Cout, difference, borrow);
2      input A, B, Cin; // Inputs for Full Adder
3      output S, Cout;   // Outputs for Full Adder
4      output difference, borrow; // Outputs for Full Subtractor
5
6      // Full Adder Logic
7      assign S = A ^ B ^ Cin;          // Sum
8      assign Cout = (A & B) | (Cin & (A ^ B)); // Carry out
9
10     // Full Subtractor Logic
11     assign difference = A ^ B ^ borrow; // Difference
12     assign borrow = (~A & B) | (borrow & ~A); // Borrow
13 endmodule
14
```

RTL Schematic:



Output Timing Waveform:



Result:

Thus the Full Adder and Full Subtractor circuits are designed and the truth tables is verified using Quartus software.

Developed by: Rhudhresh

Register Number: 212223050039