

Experiment--03-Half-Subtractor-and-Full-subtractor

› **Implementation-of-Half-subtractor-and-Full-subtractor-circuit**

› **AIM:**

To design a half subtractor 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**

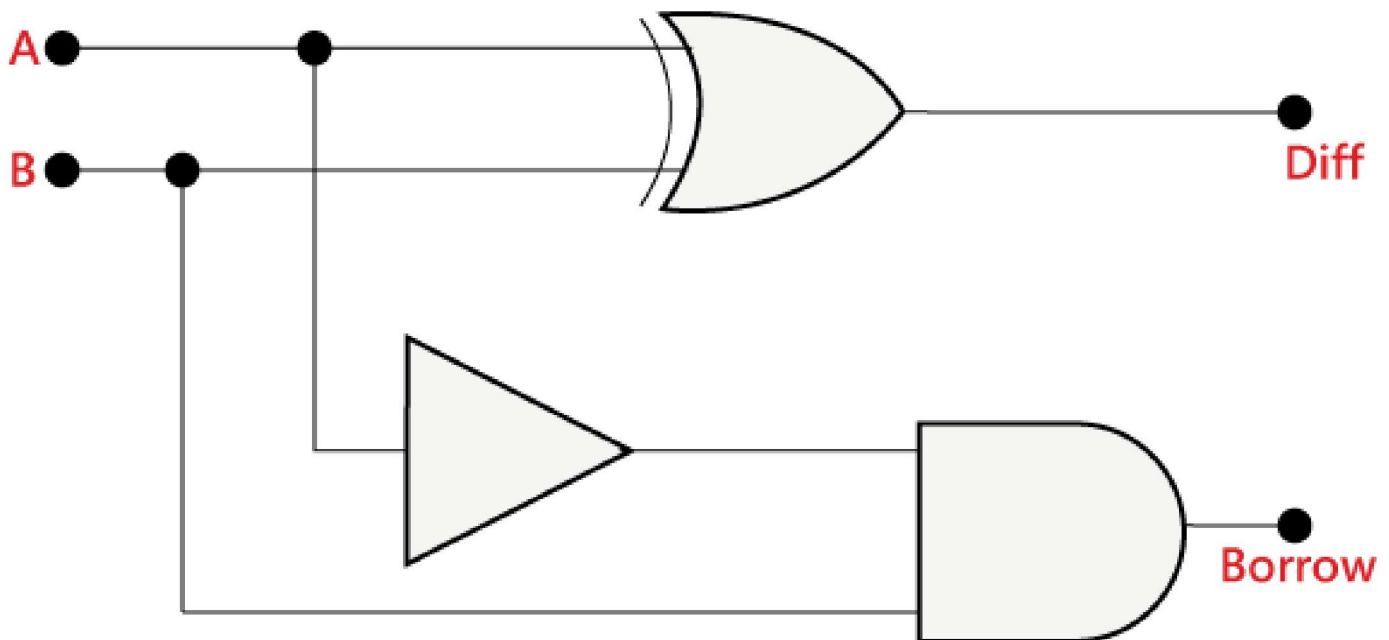
› **Theory:**

Subtractor circuits take two binary numbers as input and subtract one binary number input from the other binary number input. Similar to adders, it gives out two outputs, difference and borrow (carry-in the case of Adder). There are two types of subtractors.

' Half Subtractor Full Subtractor

' Half Subtractor:

The half-subtractor is a combinational circuit which is used to perform subtraction of two bits. It has two inputs, X (minuend) and Y (subtrahend) and two outputs D (difference) and B (borrow). To perform $x - y$, we have to check the relative magnitudes of x and y . If $x \geq y$, we have three possibilities: $0 - 0 = 0$, $1 - 0 = 1$, and $1 - 1 = 0$. The result is called the difference bit. If $x < y$, we have $0 - 1$, and it is necessary to borrow a 1 from the next higher stage. The 1 borrowed from the next higher stage adds 2 to the minuend bit, just as in the decimal system a borrow adds 10 to a minuend digit. With the minuend equal to 2, the difference becomes $2 - 1 = 1$. The half-subtractor needs two outputs. One output generates the difference and will be designated by the symbol D. The second output, designated B for borrow, generates the binary signal that informs the next stage that a 1 has been borrowed.

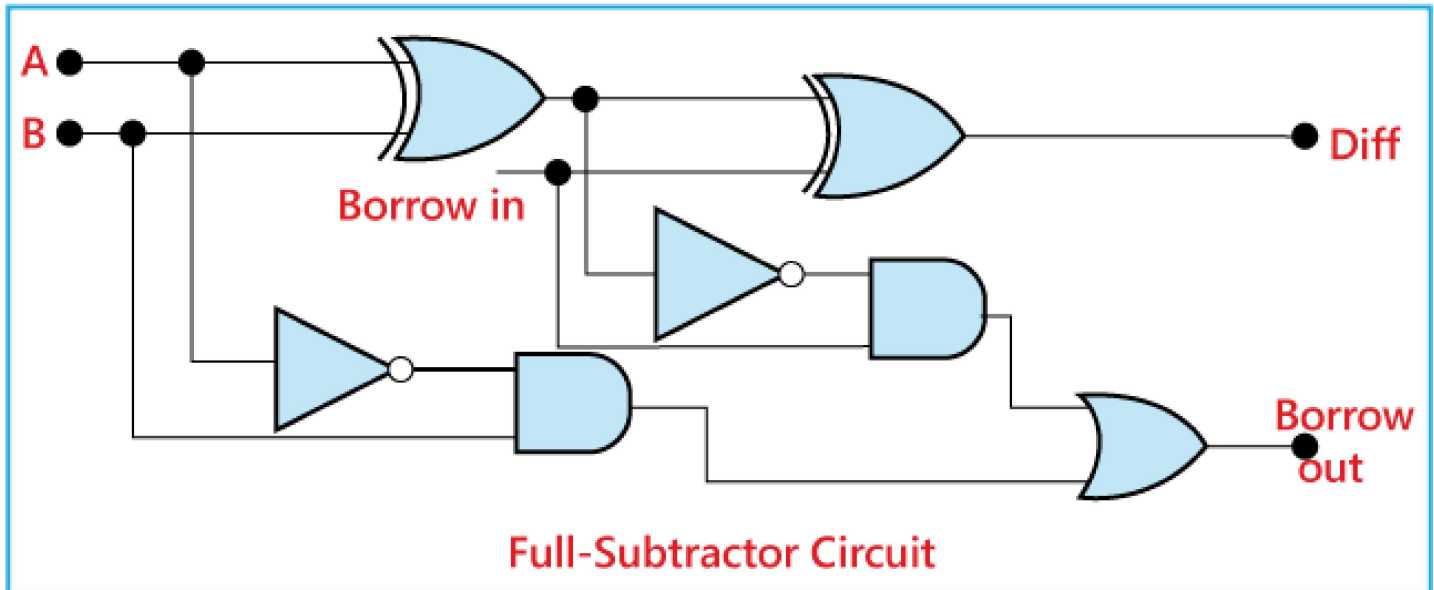


Half-Subtractor Circuit

$$\text{Sum} = X'Y + XY' = X \oplus Y \quad \text{Carry} = X'Y$$

› 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.



$$\text{Diff} = A \oplus B \oplus \text{Bin} \quad B = A' \text{Bin} + A'B + B \text{Bin}$$

› Procedure:

1. Use module projname(input,output) to start the Verilog programming.
2. Assign inputs and outputs using the word input and output respectively.
3. Use defined keywords like wire,assign and required logic gates to represent the boolean expression.
4. Use each output to represent one for difference and the other for borrow.
5. End the verilog program using keyword endmodule.

› Program:

```
/*
```

```
Program to design a half subtractor and full subtractor circuit and verify its
```

Developed by: Mohammed Faizal.J

RegisterNumber: 212222100027

*/

Half Subtractor:

```
module HalfSubtractor(a,b,diff,borr);
```

```
input a,b;
```

```
output diff,borr;
```

```
assign diff = (a^b);
```

```
assign borr = (~a&b);
```

```
endmodule
```

Full Subtractor:

```
module FullSubtractor(a,b,c,diff,borr);
```

```
input a,b,c;
```

```
output diff,borr;
```

```
assign borr = (~a&(b^c)|(b&c));
```

```
assign diff = (a^b^c);
```

```
endmodule
```



’ Output:

’ Truthtable

Half Subtractor:

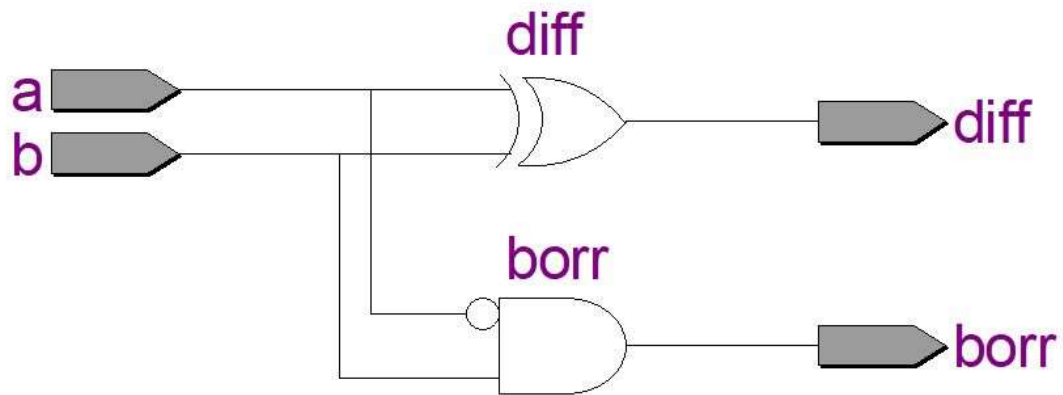
Inputs		Outputs	
A	B	Diff	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Full Subtractor:

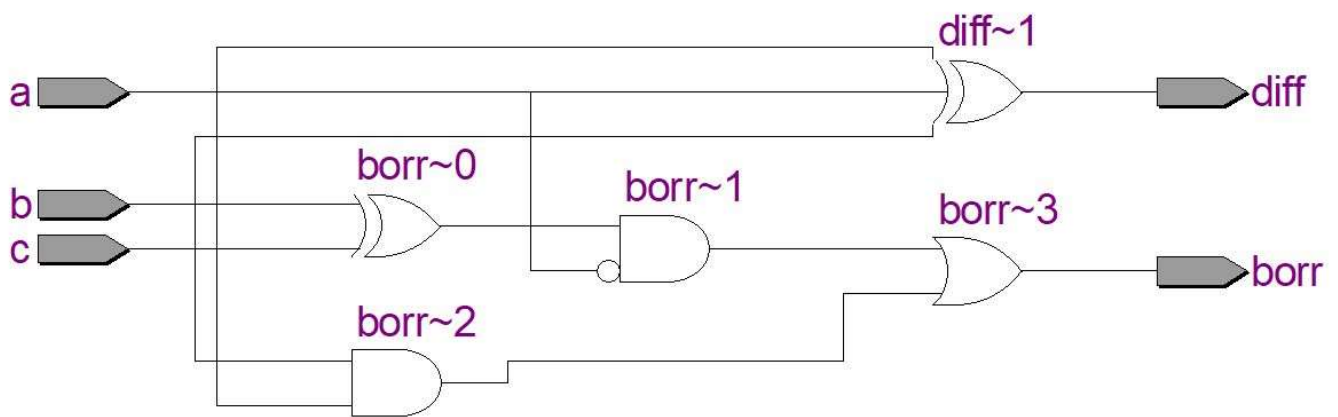
Inputs			Outputs	
A	B	Borrow _{in}	Diff	Borrow
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

’ RTL realization

’ Half Subtractor:

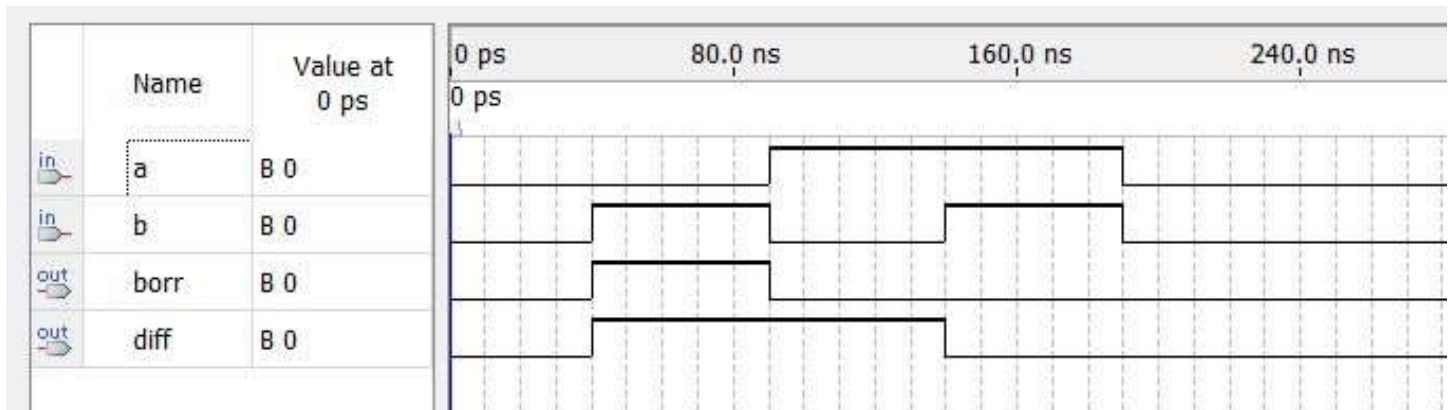


’ Full Subtractor:

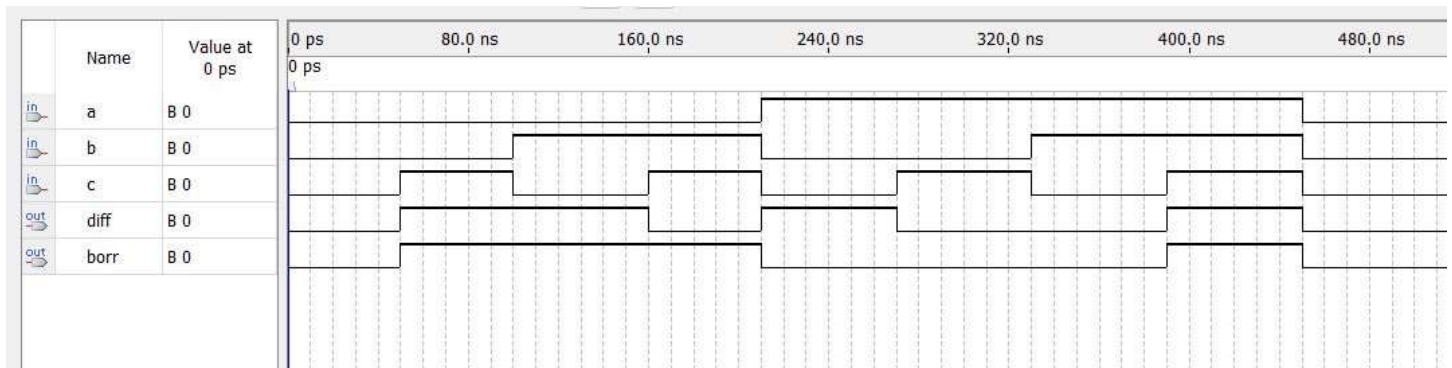


’ Timing diagram

’ Half Subtractor:



Full Subtractor:



Result:

Thus the half subtractor and full subtractor circuits are designed and the truth tables is verified using quartus software.