ganeshprabhu2005 / **HALF_ADDER_SUBTRACTOR** Public

forked from naavaneetha/HALF_ADDER_SUBTRACTOR

⚖ GPL-3.0 license

☆ **0** stars   ⑂ **392** forks   ⑂ Branches   🏷 Tags ∿ Activity

☆ Star ▾

🔔 Notifications

<> **Code**   ⑂ Pull requests   ▷ Actions   ▦ Projects   ⊘ Security   ∿ Insights

main ▾ | ⑂ **1** Branch   ◊ **0** Tags | ⑂   ◊ | 🔍 Go to file | Go to file | Code | •••

This branch is **7 commits ahead of** `naavaneetha/HALF_ADDER_SUBTRACTOR:main` .

🦜 **ganeshprabhu2005** Update README.md   1 minute ago   •••   🕐

📁 db	lab experiment	3 months ago
📁 incremental_db	lab experiment	3 months ago
📁 output_files	lab experiment	3 months ago
📁 simulation	lab experiment	3 months ago
📄 HALF_ADDSUB.qpf	lab experiment	3 months ago
📄 HALF_ADDSUB.qsf	lab experiment	3 months ago
📄 HALF_ADDSUB.qws	lab experiment	3 months ago
📄 HALF_ADDSUB.v	lab experiment	3 months ago
📄 HALF_ADDSUB.v.bak	lab experiment	3 months ago
📄 LICENSE	Initial commit	3 months ago
📄 README.md	Update README.md	1 minute ago
📄 Waveform.vwf	lab experiment	3 months ago
📄 Waveform1.vwf	lab experiment	3 months ago

📖 **README**   ⚖ License

HALF_ADDER_SUBTRACTOR

Implementation-of-Half-Adder-and-Half Subtractor-circuit

AIM:

To design a half adder and half subtractor circuit and verify its truth table in Quartus using Verilog programming.

Equipments Required:

Hardware – PCs, Cyclone II , USB flasher

Software – Quartus prime Theory Adders are digital circuits that carry out the addition of numbers.

Half Adder

Half adder is a combinational circuit that performs simple addition of two binary numbers. The input variables designate the augend and addend bits; the output variables produce the sum and carry. It is necessary to specify two output variables because the result may consist of two binary digits.
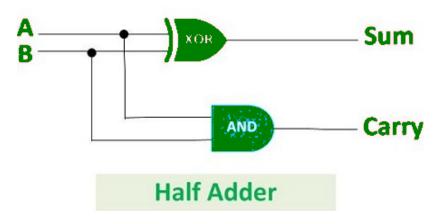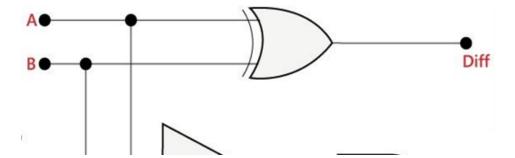
Sum = A'B+AB' =A $\oplus$ B Carry = AB

Figure -01 HALF ADDER

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 ;;, y, we have three possibilities: 0 - 0 = 0, 1 - 0 = 1, and 1 - I = 0. The result is called the difference bit. If x < y, we have 0 - I, and it is necessary to borrow a 1 from the next higher stage. The I 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 - I = 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 I has been borrowed.

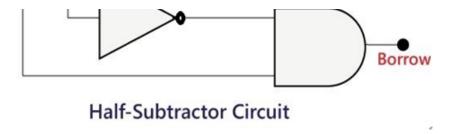Diff = A'B+AB' =A $\oplus$ B Borrow = A'B

Half-Subtractor Circuit

Figure -02 HALF Subtractor

Truthtable

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.

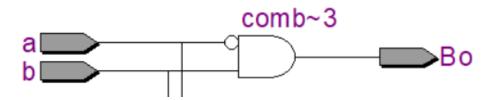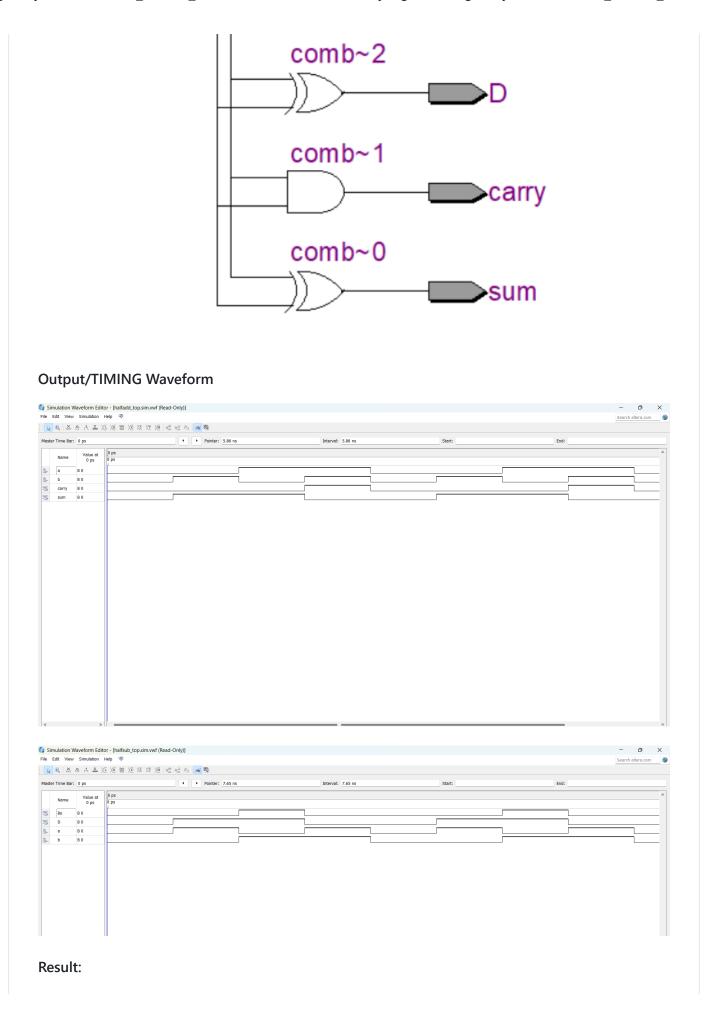5. For different input combinations generate the timing diagram.

Program:

```
module HALF_ADDSUB(a,b,sum,carry,D,Bo);
input a,b;
output sum,carry,D,Bo;
//HALF ADDER
xor(sum,a,b);
and(carry,a,b);


//HALF SUBTRACTOR
wire abar;
not(abar,a);
xor(D,a,b);
and(Bo,abar,b);
endmodule
```

Developed by: GANESH PRABHU J RegisterNumber: 212223220023

RTL Schematic

Output/TIMING Waveform

Result:

Thus Implementation-of-Half-Adder-and-Half Subtractor-circuit is running successfully

Releases

No releases published

Packages

No packages published

Languages

- **VHDL** 48.1%
- **Verilog** 17.8%
- **Stata** 16.7%
- **HTML** 16.0%
- **Standard ML** 1.4%