

# RTL CHALLENGE

**DAY-7:-** Verilog Code for Implementation of HALF SUBTRACTOR  
using GateLevel, DataFlow & Structural.

**Software used:-** Xilinx Vivado(2023.1)

## Theory:

### Half Subtractor:-

A **half-subtractor** is a combinational logic circuit that have two inputs and two outputs (i.e. difference and borrow). The half subtractor produces the difference between the two binary bits at the input and also produces a borrow output (if any). In the subtraction ( $A - B$ ), A is called as **Minuend bit** and B is called as **Subtrahend bit**. The block diagram and logic circuit diagram of the half subtractor is shown in Figure-1.

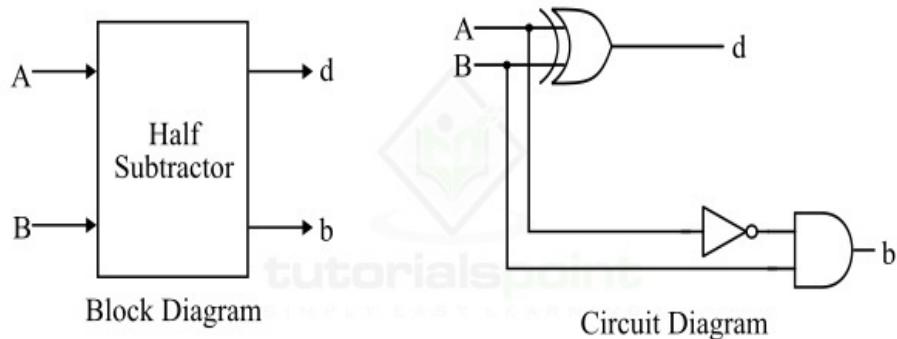


Figure 1 - Half Subtractor

## TRUTH TABLE:

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

## DESIGN CODES & TESTBENCHES:-

### Code for HalfSubtractor using GATELEVEL:

```
// Design Name: GateLevel
// Module Name: HalfSubtractor
// Project Name: RTL Challenge

module HalfSubtractorGateLevel(
    input a,b,
    output difference,borrow
);
    xor (difference,a,b);
    and (borrow,!a,b);
endmodule
```

### TESTBENCH:

```
// Design Name: TestBench
// Module Name: HalfSubtractor_tb
// Project Name: RTL Challenge

module HalfSubtractorGateLevel_tb();
reg a,b;
wire difference,borrow;
HalfSubtractorGateLevel uut (.a(a),.b(b),.difference(difference),.borrow(borrow));
initial
begin
    a=0;b=0:#100;
    a=0;b=1:#100;
    a=1;b=0:#100;
    a=1;b=1:#100;
end
endmodule
```

## Code for HalfSubtractor using DATAFLOW:

```
// Design Name: Data Flow
// Module Name: HalfSubtractorDataFlow
// Project Name: RTL Challenge

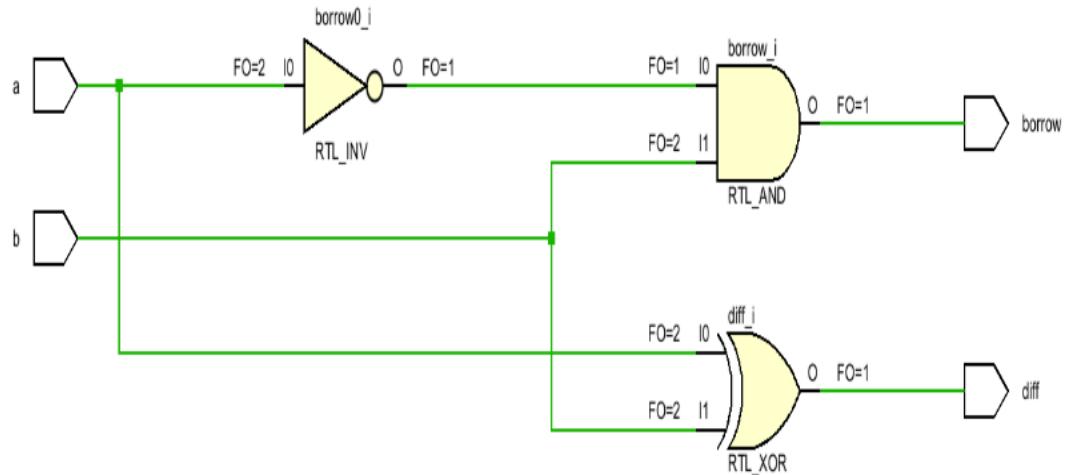
module HalfSubtractorDataFlow(
    input a,b,
    output difference,borrow
);
    assign difference = a^b;
    assign borrow = (!a&b);
endmodule
```

## TESTBENCH:

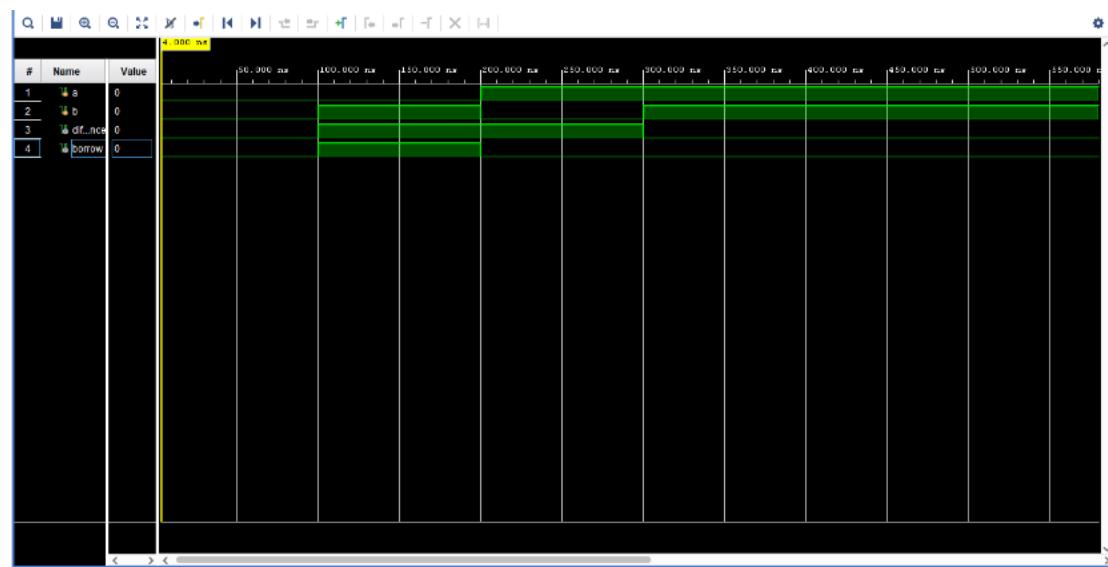
```
// Design Name: Data Flow
// Module Name: HalfSubtractorDataFlow_tb
// Project Name: RTL Challenge

module HalfSubtractorDataFlow_tb();
reg a,b;
wire difference,borrow;
HalfSubtractorDataFlow uut (.a(a),.b(b),.difference(difference),.borrow(borrow));
initial
begin
    a=0;b=0;#100;
    a=0;b=1;#100;
    a=1;b=0;#100;
    a=1;b=1;#100;
end
endmodule
```

## RTL SCHEMATIC FOR GATELEVEL & DATAFLOW:



## OUTPUT WAVEFORMS FOR GATELEVEL & DATAFLOW:



## Code for HalfSubtractor using STRUCTURAL:

```
// Design Name: STRUCTURAL
// Module Name: HalfAdderStructural
// Project Name: RTL Challenge

module HalfAdderStructural(
    input a,b,
    output difference,borrow
);

    xorgate xl (.a(a),.b(b),.y(difference));
    andgate al (.a(!a),.b(b),.y(borrow));

endmodule
```

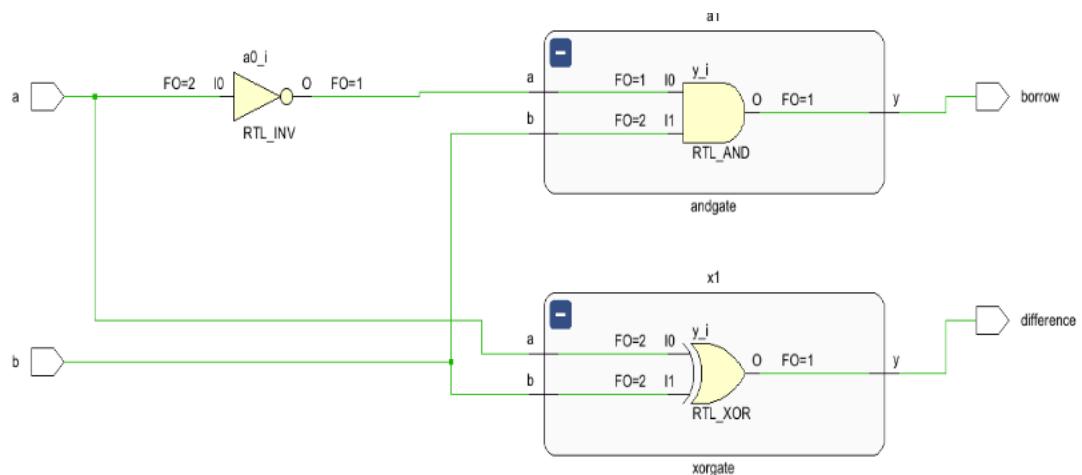
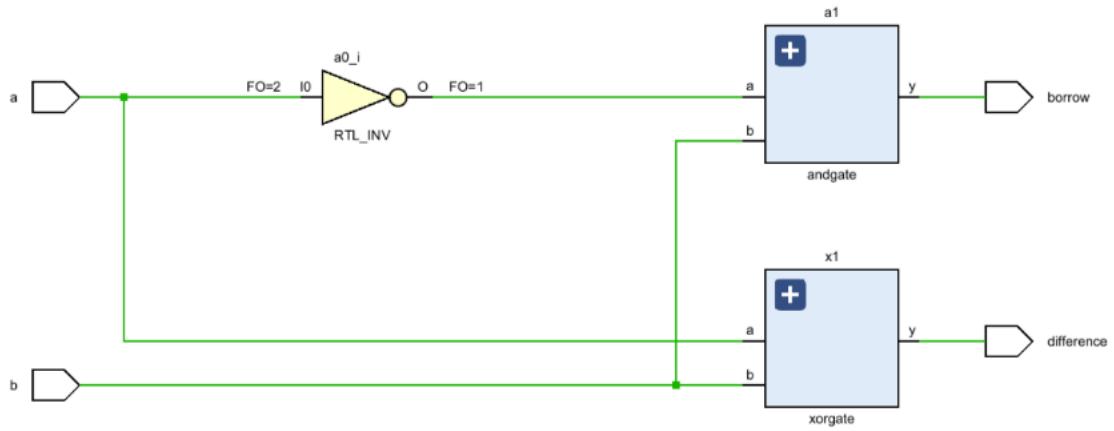
## TESTBENCH:

```
// Design Name: STRUCTURAL
// Module Name: HalfAdderStructural_tb
// Project Name: TESTBENCH

module HalfAdderStructural_tb();
reg a,b;
wire difference,borrow;
HalfAdderStructural uut(.a(a),.b(b),.difference(difference),.borrow(borrow));
initial
begin
    a=0;b=0;#100;
    a=0;b=1;#100;
    a=1;b=0;#100;
    a=01;b=1;#100;

    end
endmodule
```

## RTL SCHEMATIC FOR STRUCTURAL:



## OUTPUT WAVEFORMS FOR STRUCTURAL:

