Full Adder

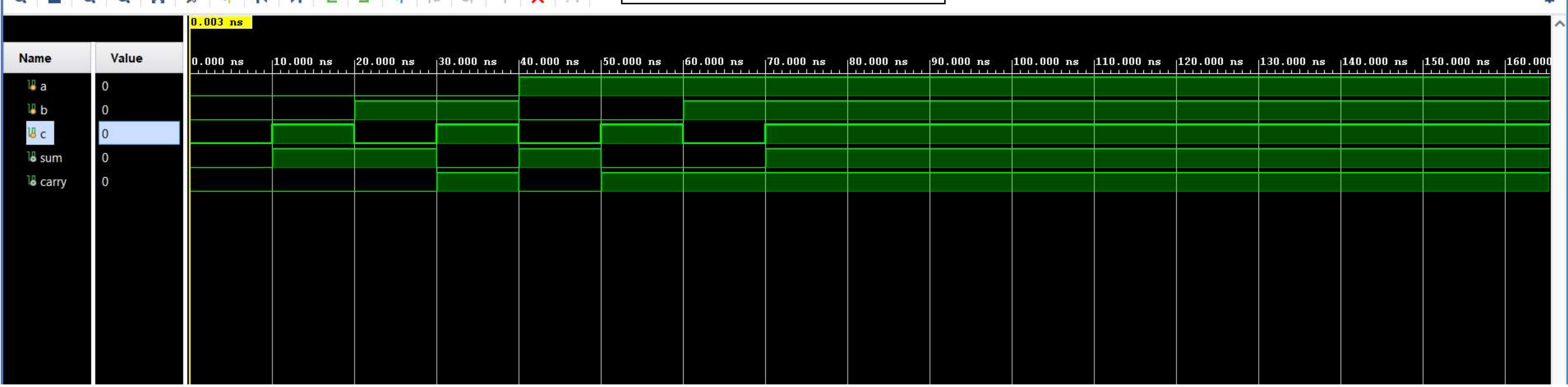
D:/Vivado/project_2/project_2.srcs/sources_1/new/Design.v

```verilog
// Code your design here
module fa(
  input a,b,c,
  output sum,carry
);
  assign sum=a^b^c;
  assign carry=(((a^b)&c)|(a&b));
endmodule
```

D:/Vivado/project_2/project_2.srcs/sim_1/new/Testbench.v

```verilog
// Code your testbench here
// or browse Examples
module fa_tb;
  reg a,b,c;
  wire sum,carry;
  fa u1(.a(a),.b(b),.c(c),.sum(sum),.carry(carry));
  initial begin
    $monitor("a=%b,b=%b,c=%b  ,sum=%b,carry=%b",a,b,c,sum,carry);
    a=1'b0;b=1'b0;c=1'b0;#10;
    a=1'b0;b=1'b0;c=1'b1;#10;
    a=1'b0;b=1'b1;c=1'b0;#10;
    a=1'b0;b=1'b1;c=1'b1;#10;
    a=1'b1;b=1'b0;c=1'b0;#10;
    a=1'b1;b=1'b0;c=1'b1;#10;
    a=1'b1;b=1'b1;c=1'b0;#10;
    a=1'b1;b=1'b1;c=1'b1;#10;
  end
endmodule
```

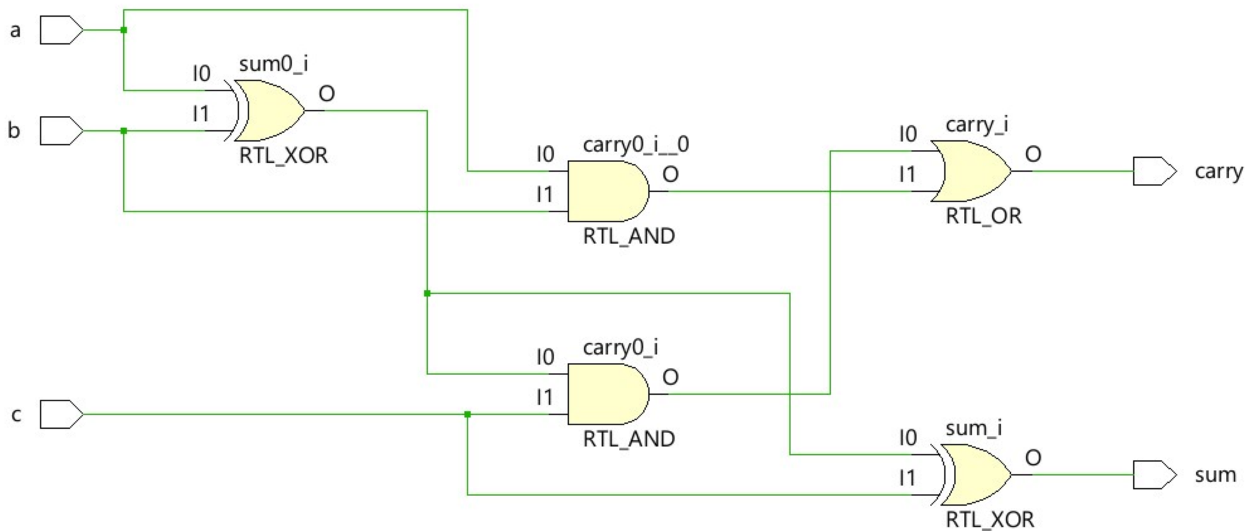
Timing Digram



Digram

OUTPUT

e




```
# KERNEL: a=0,b=0,c=0  ,sum=0,carry=0
# KERNEL: a=0,b=0,c=1  ,sum=1,carry=0
# KERNEL: a=0,b=1,c=0  ,sum=1,carry=0
# KERNEL: a=0,b=1,c=1  ,sum=0,carry=1
# KERNEL: a=1,b=0,c=0  ,sum=1,carry=0
# KERNEL: a=1,b=0,c=1  ,sum=0,carry=1
# KERNEL: a=1,b=1,c=0  ,sum=0,carry=1
# KERNEL: a=1,b=1,c=1  ,sum=1,carry=1
```

A **Full Adder** is a fundamental combinational logic circuit in digital electronics that adds three binary inputs and produces two outputs: a sum and a carry125.

Working Principle

Unlike a half adder, which can only add two bits, a full adder can add three bits at a time: two significant bits (A and B) and a carry-in (Cin) from a previous addition. This makes full adders suitable for multi-bit binary addition by cascading several full adders together1235.

Truth Table

A	B	Cin	Sum (S)	Carry-out (Cout)
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

Boolean Expressions

- **Sum:**
  S=A⊕B⊕Cin$S=A\oplus B\oplus Cin$
  (XOR of all three inputs)

- **Carry-out:**
  $Cout=(A \cdot B)+(B \cdot Cin)+(A \cdot Cin)$
  (Carry is 1 if at least two inputs are 1)

Circuit Implementation

A full adder can be constructed using:

- Two XOR gates (for the sum)
- Three AND gates and one OR gate (for the carry)[14]

Alternatively, a full adder can be built by combining two half adders and an OR gate:

- First half adder adds A and B.
- Second half adder adds the sum from the first half adder to Cin.
- Carry outputs from both half adders are combined with an OR gate to produce Cout[25].

Applications

- Used in arithmetic logic units (ALUs) of microprocessors
- Essential for multi-bit binary addition (by cascading multiple full adders)
- Core component in digital calculators, computers, and other digital systems[6]

Advantages Over Half Adder

- Can process carry-in, enabling multi-bit binary addition
- Consumes less power and operates at higher speed compared to half adders for larger operations

- Easily cascaded for n-bit addition[124]

In summary:
A full adder is a digital circuit that adds three binary inputs (A, B, Cin) and produces a sum and a carry-out, making it a key building block for complex arithmetic operations in digital systems