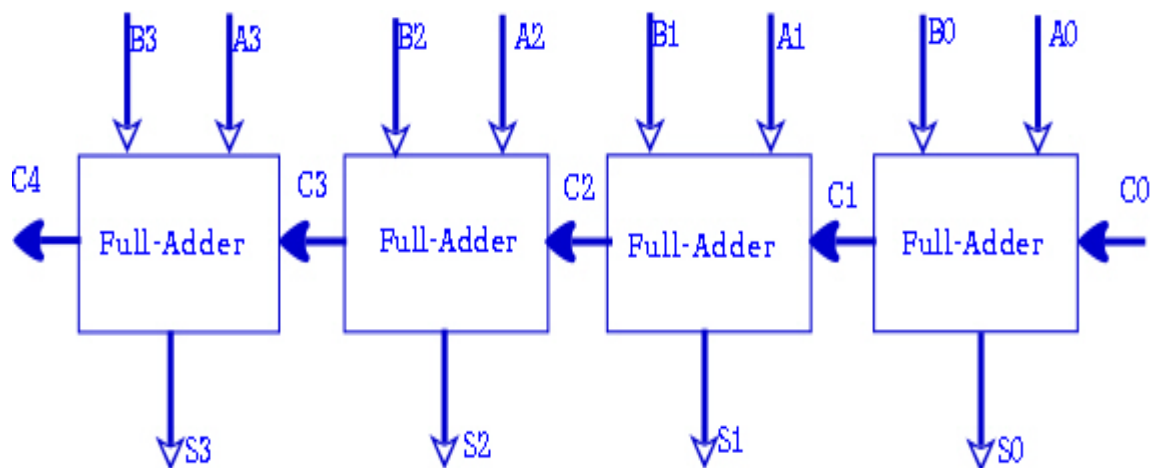


## 4-bit adder

### Introduction:

A 4-bit binary adder is a fundamental electronic circuit used to do addition operations. The adder takes two 4-bit inputs, A and B to produce a 4-bit output, sum along with a carry out. This means you can calculate  $1111+1111$ , which will give you  $11110$  ( $15+15 = 30$ ).

### Circuit:



## Code:

### Mux.v

```
module Four_bit_adder(a,b,c_in,s,c_out);
input [3:0]a,b;
input c_in;
output [3:0]s;
output c_out;
wire c1,c2,c3;

Full_adder H0(a[0],b[0],c_in,s[0],c1);
Full_adder H1(a[1],b[1],c1,s[1],c2);
Full_adder H2(a[2],b[2],c2,s[2],c3);
Full_adder H3(a[3],b[3],c3,s[3],c_out);
endmodule

////////////////////////////////////
module Full_adder(a,b,c_in,s,c);
input a,b,c_in;
output s,c;
wire x,y,z;
xor(x,a,b);
xor(s,x,c_in);
and(y,a,b);
and(z,x,c_in);
xor(c,z,y);
endmodule
```

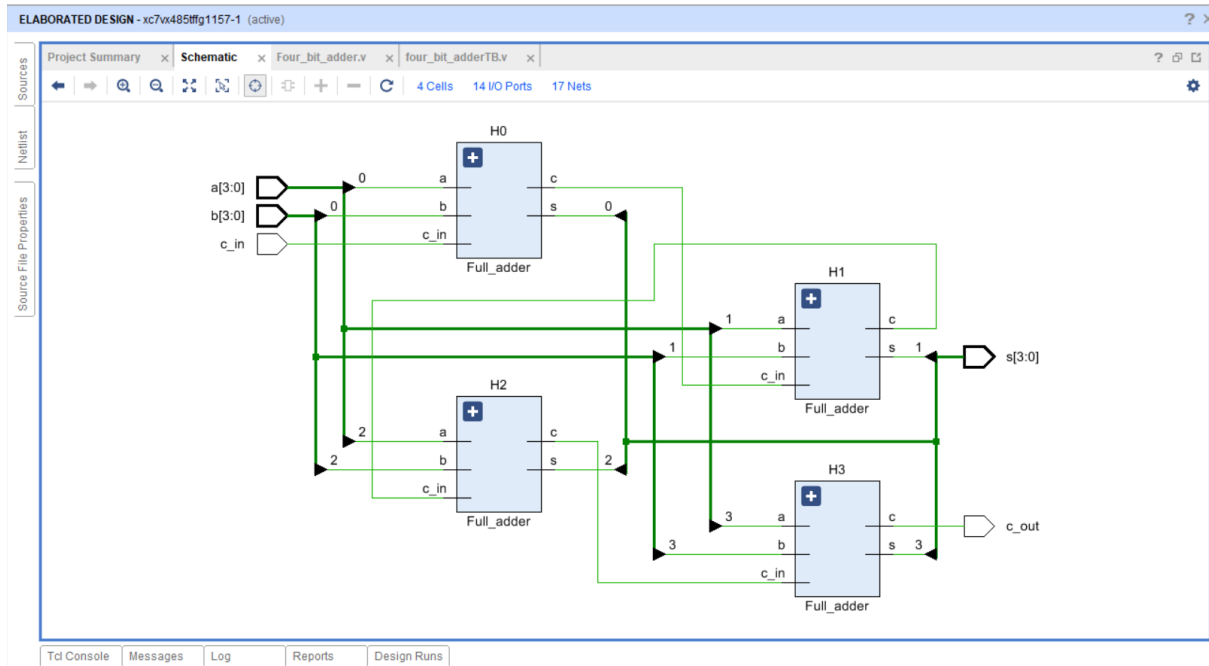
### Testbench

```
module four_bit_adderTB();
reg [3:0]A,B;
reg C_in;
wire [3:0]S;
wire C_out;
Four_bit_adder T2(A,B,C_in,S,C_out);
initial
begin
A=4'd8;B=4'd9;C_in=1'b1;
#100
$display("Time=%0d A=%d B=%d C_in=%d
S=%d C_out=%b\n", $time,A, B ,C_in ,S ,C_out);
A=4'd2;B=4'd7;C_in=1'b0;
#100
$display("Time=%0d A=%d B=%d C_in=%d S=%d
C_out=%b\n",$time,A, B ,C_in ,S ,C_out);
A=4'd9;B=4'd2;C_in=1'b0;
#100
$display("Time=%0d A=%d B=%d C_in=%d S=%d
C_out=%b\n",$time,A, B ,C_in ,S ,C_out);
A=4'd9;B=4'd2;C_in=1'b0;
end
endmodule
```

## Simulation:



## Elaborate design:



## TCL console:

```
Time=100  A= 8  B= 9  C_in=1  S= 2  C_out=1
```

```
Time=200 A= 2  B= 7  C_in=0  S= 9  C_out=0
```

```
Time=300 A= 9  B= 2  C_in=0  S=11  C_out=0
```