FULL ADDER

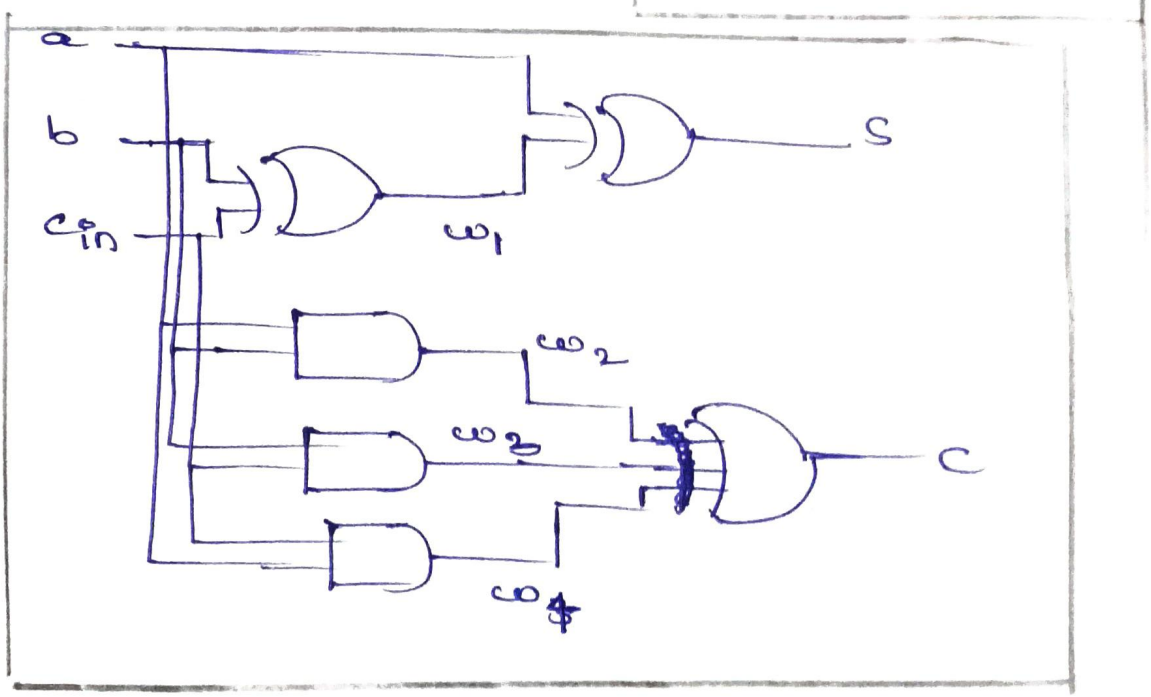
Design code:-

```
module FA (
 input a, b, cin,
 output S, C
 );
 wire w1, w2, w3, w4;
 xor (w1, b, cin);
 xor (S, w1, a);
 and (w2, a, b);
 and (w3, b, cin);
 and (w4, cin, a);
 or (C, w1, w2, w3);
endmodule
```

a	b	c_{in}	S	C
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

$S = a \oplus (b \oplus c_{in})$

$C = ab + bc_{in} + c_{in}a$



```
module FA(
input a, b, cin,
output S, C,
);
assign   S = a ^ b ^ cin;
assign   c = (a & b) | (b & cin) | (cin & a);
endmodule
```

Testbench :-

```
module FA_tb( );
reg a, b, cin;
wire S, C;
FA uut (a, b, cin, S, C);
initial begin
  a = 0; b = 1; cin = 1;
  #10
  a = 1; b = 1; cin = 1;
  #10
  a = 1; b = 0; cin = 0;
  #10
  a = 0; b = 0; cin = 0;
  end
  $finish( );
  endmodule
```