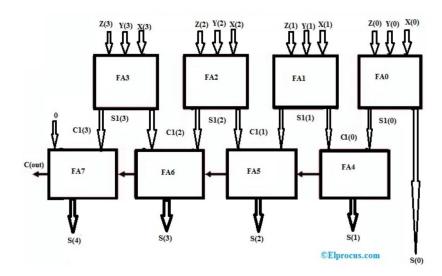


DAY-13 #100DAYSOFRTL

Aim: Implementation of 4-Bit Carry Save Adder(CSA).

Block Diagram:-



RTL CODE:-

```
////DATE:-13/01/2024
   .
!////CARRY SAVE ADDER using FULL ADDERS
   module FULLadder (input A, B, Cin,
   output Sum, Carry);
O assign Sum=A^B^Cin;
O assign Carry=A&B|B&Cin|Cin&A;
   endmodule
   ////Carry Save Adder
   module CSA(input [3:0] A,B,C,
   output [5:0] Sum,
   output Cout);
   wire [3:0] S;
   wire [3:0] C1;
   wire [3:0] C2;
   FULLadder M1(.A(A[0]),.B(B[0]),.Cin(C[0]),.Sum(S[0]),.Carry(C1[0]));
   FULLadder M2(.A(A[1]),.B(B[1]),.Cin(C[1]),.Sum(S[1]),.Carry(C1[1]));
   FULLadder M3(.A(A[2]),.B(B[2]),.Cin(C[2]),.Sum(S[2]),.Carry(C1[2]));
   FULLadder M4(.A(A[3]),.B(B[3]),.Cin(C[3]),.Sum(S[3]),.Carry(C1[3]));
   FULLadder M5(.A(C1[0]),.B(S[1]),.Cin(0),.Sum(Sum[1]),.Carry(C2[0]));
   FULLadder M6(.A(C1[1]),.B(S[2]),.Cin(C2[0]),.Sum(Sum[2]),.Carry(C2[1]));
   FULLadder M7(.A(C1[2]),.B(S[3]),.Cin(C2[1]),.Sum(Sum[3]),.Carry(C2[2]));
   FULLadder M8(.A(C1[3]),.B(0),.Cin(C2[2]),.Sum(Sum[4]),.Carry(Cout));
o |assign Sum[0]=S[0];
O assign Sum[5]=Cout;
   endmodule
```

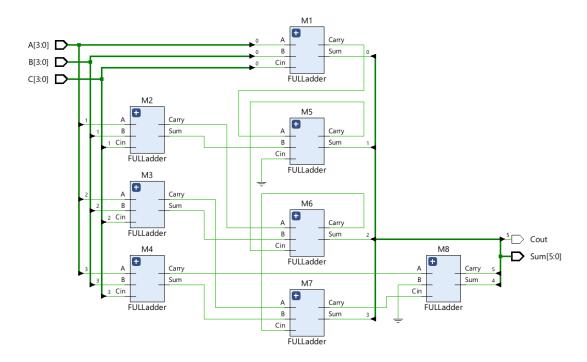
TESTBENCH:-

```
module CSA tb();
   reg [3:0] A,B,C;
   wire [5:0] Sum;
   wire Cout;
   CSA dut(.A(A),.B(B),.C(C),.Sum(Sum),.Cout(Cout));
   initial begin
O A=4'b0011; B=4'b0100;
O C=4'b1000;
O #10;
O A=4'b0001; B=4'b0010;
O c=4'b0010;
O !#10;
O |$display("A=%d,B=%d,C=%d,Cout=%b,Sum=%d",A,B,C,Cout,Sum);
O A=4'b0011; B=4'b1000;
O C=4'b0100;
O !#10;
$\$\display(\"A=\%d,B=\%d,C=\%d,Cout=\%b,Sum=\%d\",A,B,C,Cout,Sum);
O A=4'b1010; B=4'b0101;
O 'C=4'b0110;
O ;#10;
$\display("A=\d,B=\d,C=\d,Cout=\d,Sum=\d",A,B,C,Cout,Sum);
O A=4'b0111; B=4'b0100;
O |C=4'b1100;
O \#10;
$\display("A=\d,B=\d,C=\d,Cout=\b,Sum=\d",A,B,C,Cout,Sum);
O A=4'b0110; B=4'b0110;
O C=4'b1000;
O #10;
O | $\frac{1}{2} \display ("A=\frac{1}{2}d, B=\frac{1}{2}d, Cout=\frac{1}{2}b, Sum=\frac{1}{2}d", A, B, C, Cout, Sum);
O 'A=4'b1111; B=4'b1111;
O |C=4'b1111;
O !#10;
$\display("A=\d,B=\d,C=\d,Cout=\b,Sum=\d",A,B,C,Cout,Sum);
○⇒$finish();
    end
    endmodule
```

OUTPUT:-

```
A= 3,B= 4,C= 8,Cout=0,Sum=15
A= 1,B= 2,C= 2,Cout=0,Sum= 5
A= 3,B= 8,C= 4,Cout=0,Sum=15
A=10,B= 5,C= 6,Cout=0,Sum=21
A= 7,B= 4,C=12,Cout=0,Sum=23
A= 6,B= 6,C= 8,Cout=0,Sum=20
A=15,B=15,C=15,Cout=1,Sum=45
```

SCHEMATIC:-



WAVEFORMS:-

Name	Value	0.000 ns	10.000 ns	20.000 ns	30.000 ns	40.000 ns	50.000 ns	60.000 n	
> W A[3:0	1111	0011	0001	0011	1010	0111	0110	1.	11
> 💆 B[3:0	1111	0100	0010	1000	0101	0100	0110	1:	11
> V C[3:0	1111	1000	0010	0100	0110	1100	1000	1:	11
> ⊌ Su(101101	001111	000101	001111	010101	010111	010100	10:	101
™ Cout	1								