Experiment 7: Bit-Serial Adder

Rahul Nimbal (2016EE30527)

AIM: To study and implement Bit-Serial Adder.

Material Required: CPLD Board.

Logic Diagram:

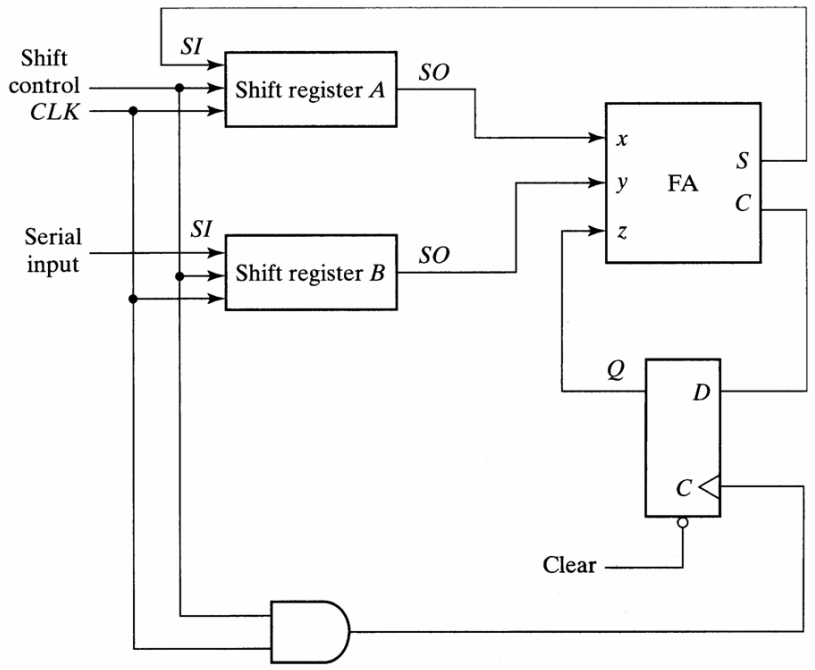

FIGURE 6-5
Serial Adder

Verilog Code:

```
module shift ( s ,inp, clock );



input inp ;
wire inp ;
input clock ;
wire clock ;
output [3:0] s;

reg [3:0]s;

always @ (posedge (clock)) begin
begin

  s[3] <= s[2] ;
  s[2] <= s[1] ;
  s[1] <= s[0] ;
  s[0] <= inp ;

 end
end

endmodule
```

```
module sa(a, sum,cout,b,clock, t,shi);
input b;
input clock;
input shi;
wire [3:0] x,z;
output [3:0] a;
output sum, cout;

output t;
wire s,cin, t;

assign t = clock & shi;
fa k(sum,cout,x[3],z[3],cin);
dff q(cin,cout,clock & shi);

shift g(x,sum,clock & shi);
shift h(z,b,clock & shi);
assign a = x;

endmodule


module fa(s,cout,a,b,cin);
input a,b,cin;
```

```
output s,cout;
assign {cout,s}=a+b+cin;
endmodule


module dff(q,d,clock);
input d,clock;
output q;
reg q;
initial begin
q=1'b0;
end
always @(posedge clock)
begin
q=d;
end
endmodule
```

Conclusion:

This experiment gave us a good insight into the syntax of Verilog code- how to implement Bit-Serial Adder. We experienced the live addition of bits with each clock cycle and pausing the shifting just anywhere using shift control.