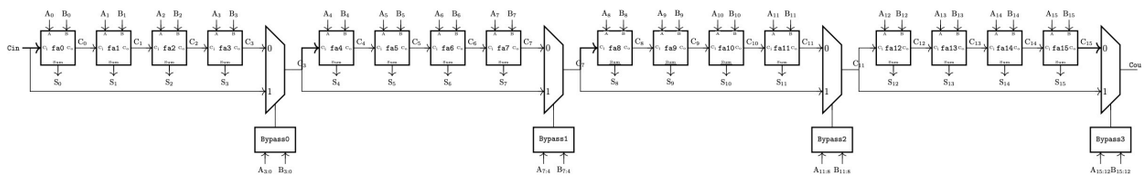
Tutorial_9 Report

Mourya Sai Sandeep EE22B045

Question_1 code:

Implement the below carry-bypass adder in verilog:




```
module carrybypass(
    input [15:0] A,
    input [15:0] B,
    input Cin,
    output Cout,
    output [15:0] S);

    wire [3:0] c;
    cbp4bit cbp0(.A(A[3:0]), .B(B[3:0]), .Cin(Cin), .Cout(c[0]), .S(S[3:0]));
    cbp4bit cbp1(.A(A[7:4]), .B(B[7:4]), .Cin(c[0]), .Cout(c[1]), .S(S[7:4]));
    cbp4bit cbp2(.A(A[11:8]), .B(B[11:8]), .Cin(c[1]), .Cout(c[2]), .S(S[11:8]));
    cbp4bit cbp3(.A(A[15:12]), .B(B[15:12]), .Cin(c[2]), .Cout(c[3]), .S(S[15:12]));

    assign Cout = c[3];
endmodule

module cbp4bit(
    input [3:0] A,
    input [3:0] B,
    input Cin,
    output Cout,
    output [3:0] S);

    wire [4:0] C;
    wire q, r;

    assign C[0] = Cin;

    fulladdr fa0(.a(A[0]), .b(B[0]), .cin(C[0]), .cout(C[1]), .Sum(S[0]));
    fulladdr fa1(.a(A[1]), .b(B[1]), .cin(C[1]), .cout(C[2]), .Sum(S[1]));
    fulladdr fa2(.a(A[2]), .b(B[2]), .cin(C[2]), .cout(C[3]), .Sum(S[2]));
    fulladdr fa3(.a(A[3]), .b(B[3]), .cin(C[3]), .cout(C[4]), .Sum(S[3]));
```

```
    bypass by0(.A(A), .B(B), .p(q));
    mux21 mu0(.i0(C[4]), .i1(C[0]), .sel(q), .o(r));

    assign Cout = r;
endmodule

module mux21(
    input i0,
    input i1,
    input sel,
    output o);

    assign o = sel ? i1 : i0;
endmodule

module bypass(
    input [3:0] A,
    input [3:0] B,
    output p);

    wire [3:0] P;
    assign P = A ^ B;
    assign p = (P[0] & P[1] & P[2] & P[3]);
endmodule

module fulladdr(
    input a,
    input b,
    input cin,
    output cout,
    output Sum);

    assign cout = (a & b) | (a & cin) | (b & cin);
    assign Sum = a ^ b ^ cin;
endmodule
```

Question_2 code:

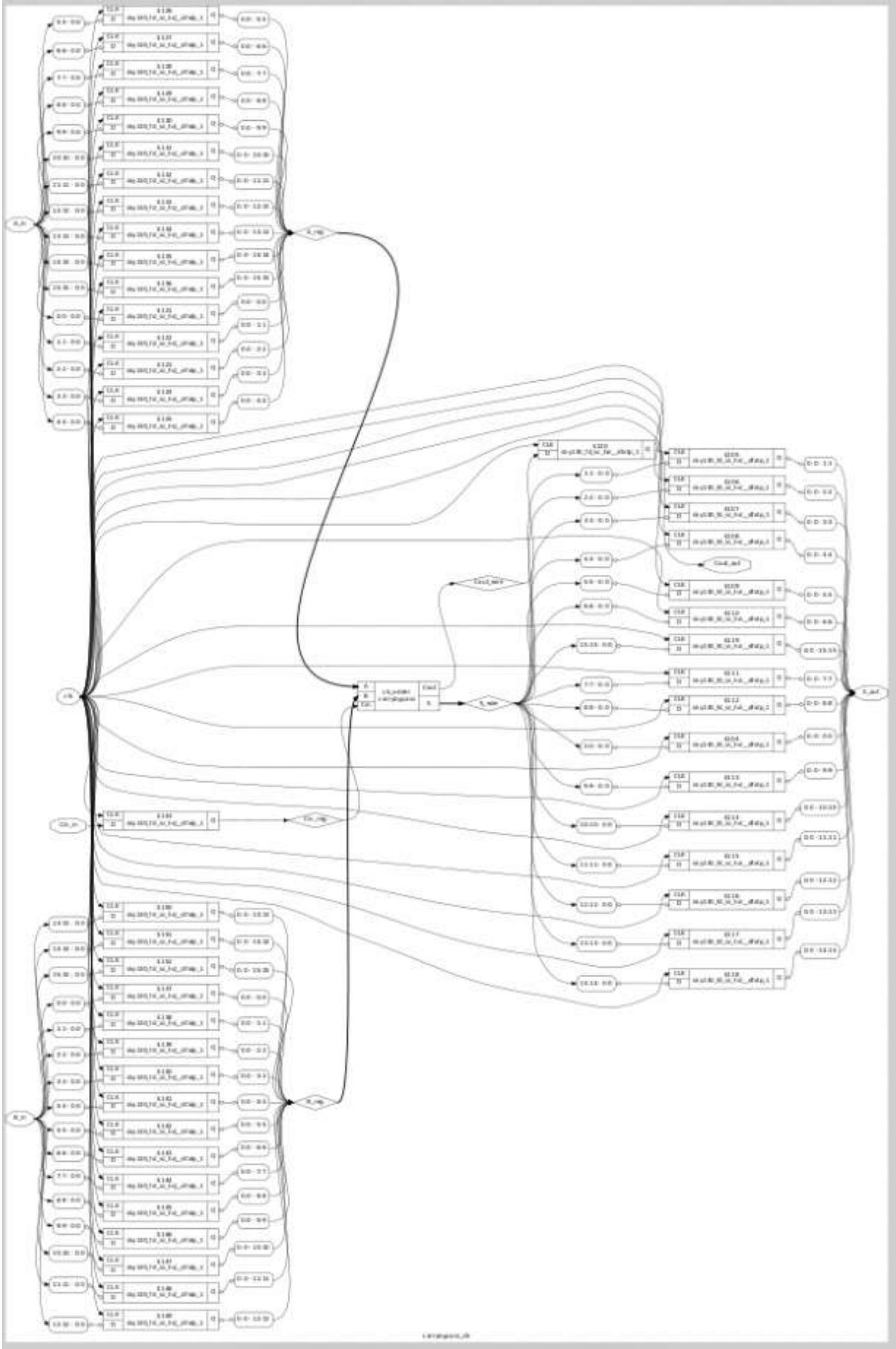
with Clock addition ~>

```
module carrybypass_clk(
    input clk,
    input [15:0] A_in,
    input [15:0] B_in,
    input Cin_in,
    output reg [15:0] S_out,
    output reg Cout_out);
```

```
    reg [15:0] A_reg;
    reg [15:0] B_reg;
    reg Cin_reg;

    wire [15:0] S_wire;
    wire Cout_wire;

    carrybypass cb_adder(
        .A(A_reg),
        .B(B_reg),
        .Cin(Cin_reg),
        .Cout(Cout_wire),
        .S(S_wire));

    always @(posedge clk) begin
        A_reg <= A_in;
        B_reg <= B_in;
        Cin_reg <= Cin_in;

        S_out <= S_wire;
        Cout_out <= Cout_wire;
    end

endmodule


module carrybypass(
    input [15:0] A,
    input [15:0] B,
    input Cin,
    output Cout,
    output [15:0] S);

    wire [3:0] c;

    cbp4bit cbp0(.A(A[3:0]),     .B(B[3:0]),     .Cin(Cin),   .Cout(c[0]), .S(S[3:0]));
    cbp4bit cbp1(.A(A[7:4]),     .B(B[7:4]),     .Cin(c[0]),  .Cout(c[1]), .S(S[7:4]));
    cbp4bit cbp2(.A(A[11:8]),    .B(B[11:8]),    .Cin(c[1]),  .Cout(c[2]), .S(S[11:8]));
    cbp4bit cbp3(.A(A[15:12]),   .B(B[15:12]),   .Cin(c[2]),  .Cout(c[3]), .S(S[15:12]));

    assign Cout = c[3];
endmodule

module cbp4bit(
    input [3:0] A, B,
    input Cin,
    output Cout,
```

```
    output [3:0] S);

    wire [4:0] C;
    wire q, r;

    assign C[0] = Cin;

    fulladdr fa0(.a(A[0]), .b(B[0]), .cin(C[0]), .cout(C[1]), .Sum(S[0]));
    fulladdr fa1(.a(A[1]), .b(B[1]), .cin(C[1]), .cout(C[2]), .Sum(S[1]));
    fulladdr fa2(.a(A[2]), .b(B[2]), .cin(C[2]), .cout(C[3]), .Sum(S[2]));
    fulladdr fa3(.a(A[3]), .b(B[3]), .cin(C[3]), .cout(C[4]), .Sum(S[3]));

    bypass by0(.A(A), .B(B), .p(q));
    mux21 mu0(.i0(C[4]), .i1(C[0]), .sel(q), .o(r));

    assign Cout = r;
endmodule

module mux21(
    input i0,
    input i1,
    input sel,
    output o);
    assign o = sel ? i1 : i0;
endmodule

module bypass(
    input [3:0] A,
    input [3:0] B,
    output p);
    wire [3:0] P;
    assign P = A ^ B;
    assign p = P[0] & P[1] & P[2] & P[3];
endmodule

module fulladdr(
    input a,
    input b,
    input cin,
    output cout,
    output Sum);
    assign cout = (a & b) | (a & cin) | (b & cin);
    assign Sum = a ^ b ^ cin;
endmodule
```

Question_3:



Tutorial_9 Report

Question_4:

```
% create_clock -name clk -period 7.5 clk
% report_checks -path_delay max
Startpoint: _17_ (rising edge-triggered flip-flop clocked by clk)
Endpoint: _16_ (rising edge-triggered flip-flop clocked by clk)
Path Group: clk
Path Type: max

  Delay    Time   Description
---------------------------------------------------------
   0.00    0.00   clock clk (rise edge)
   0.00    0.00   clock network delay (ideal)
   0.00    0.00 ^ _17_/CLK (sky130_fd_sc_hd__dfxtp_1)
   0.30    0.30 v _17_/Q (sky130_fd_sc_hd__dfxtp_1)
   0.37    0.67 v cb_adder/cbp0/fa0/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    1.06 v cb_adder/cbp0/fa1/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    1.44 v cb_adder/cbp0/fa2/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.34    1.78 v cb_adder/cbp0/fa3/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.32    2.10 v cb_adder/cbp0/mu0/_4_/X (sky130_fd_sc_hd__mux2_1)
   0.38    2.48 v cb_adder/cbp1/fa0/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    2.87 v cb_adder/cbp1/fa1/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    3.25 v cb_adder/cbp1/fa2/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.34    3.60 v cb_adder/cbp1/fa3/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.32    3.91 v cb_adder/cbp1/mu0/_4_/X (sky130_fd_sc_hd__mux2_1)
   0.38    4.30 v cb_adder/cbp2/fa0/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    4.68 v cb_adder/cbp2/fa1/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    5.06 v cb_adder/cbp2/fa2/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.34    5.41 v cb_adder/cbp2/fa3/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.32    5.73 v cb_adder/cbp2/mu0/_4_/X (sky130_fd_sc_hd__mux2_1)
   0.38    6.11 v cb_adder/cbp3/fa0/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    6.49 v cb_adder/cbp3/fa1/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    6.88 v cb_adder/cbp3/fa2/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.34    7.22 v cb_adder/cbp3/fa3/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.27    7.49 v cb_adder/cbp3/mu0/_4_/X (sky130_fd_sc_hd__mux2_1)
   0.00    7.49 v _16_/D (sky130_fd_sc_hd__dfxtp_1)
           7.49   data arrival time

   7.50    7.50   clock clk (rise edge)
   0.00    7.50   clock network delay (ideal)
   0.00    7.50   clock reconvergence pessimism
           7.50 ^ _16_/CLK (sky130_fd_sc_hd__dfxtp_1)
  -0.12    7.38   library setup time
           7.38   data required time
```

```
-----------------------------------------------------------
         7.38   data required time
        -7.49   data arrival time
-----------------------------------------------------------
        -0.12   slack (VIOLATED)
```

```
Critical Paths ~> cb_adder/cbp1/fa0/_10_/X
                  cb_adder/cbp1/fa1/_10_/X
                  cb_adder/cbp1/fa2/_10_/X
                  cb_adder/cbp1/fa3/_10_/X
```

Slack ~> -0.12 (VIOLATED)

Question_5:

```
% create_clock -name clk -period 7.5 clk
% set_false_path -through
% report_checks -path_delay max
Startpoint: _17_ (rising edge-triggered flip-flop clocked by clk)
Endpoint: _16_ (rising edge-triggered flip-flop clocked by clk)
Path Group: clk
Path Type: max

  Delay    Time   Description
---------------------------------------------------------
   0.00    0.00   clock clk (rise edge)
   0.00    0.00   clock network delay (ideal)
   0.00    0.00 ^ _17_/CLK (sky130_fd_sc_hd__dfxtp_1)
   0.30    0.30 v _17_/Q (sky130_fd_sc_hd__dfxtp_1)
   0.37    0.67 v cb_adder/cbp0/fa0/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    1.06 v cb_adder/cbp0/fa1/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    1.44 v cb_adder/cbp0/fa2/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.34    1.78 v cb_adder/cbp0/fa3/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.32    2.10 v cb_adder/cbp0/mu0/_4_/X (sky130_fd_sc_hd__mux2_1)
   0.38    2.48 v cb_adder/cbp1/fa0/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    2.87 v cb_adder/cbp1/fa1/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    3.25 v cb_adder/cbp1/fa2/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.34    3.60 v cb_adder/cbp1/fa3/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.32    3.91 v cb_adder/cbp1/mu0/_4_/X (sky130_fd_sc_hd__mux2_1)
   0.38    4.30 v cb_adder/cbp2/fa0/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    4.68 v cb_adder/cbp2/fa1/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    5.06 v cb_adder/cbp2/fa2/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.34    5.41 v cb_adder/cbp2/fa3/_10_/X (sky130_fd_sc_hd__maj3_1)
```

```
   0.32    5.73 v cb_adder/cbp2/mu0/_4_/X (sky130_fd_sc_hd__mux2_1)
   0.38    6.11 v cb_adder/cbp3/fa0/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    6.49 v cb_adder/cbp3/fa1/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.38    6.88 v cb_adder/cbp3/fa2/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.34    7.22 v cb_adder/cbp3/fa3/_10_/X (sky130_fd_sc_hd__maj3_1)
   0.27    7.49 v cb_adder/cbp3/mu0/_4_/X (sky130_fd_sc_hd__mux2_1)
   0.00    7.49 v _16_/D (sky130_fd_sc_hd__dfxtp_1)
           7.49   data arrival time

   7.50    7.50   clock clk (rise edge)
   0.00    7.50   clock network delay (ideal)
   0.00    7.50   clock reconvergence pessimism
           7.50 ^ _16_/CLK (sky130_fd_sc_hd__dfxtp_1)
  -0.12    7.38   library setup time
           7.38   data required time
-----------------------------------------------------------
           7.38   data required time
          -7.18   data arrival time
-----------------------------------------------------------
           0.20   slack (MIL)
```