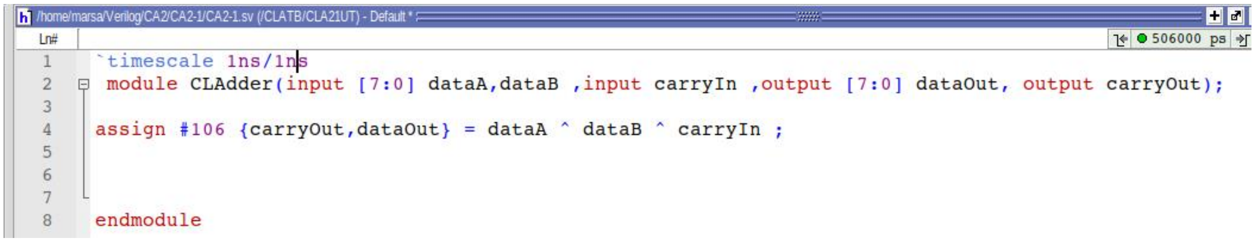
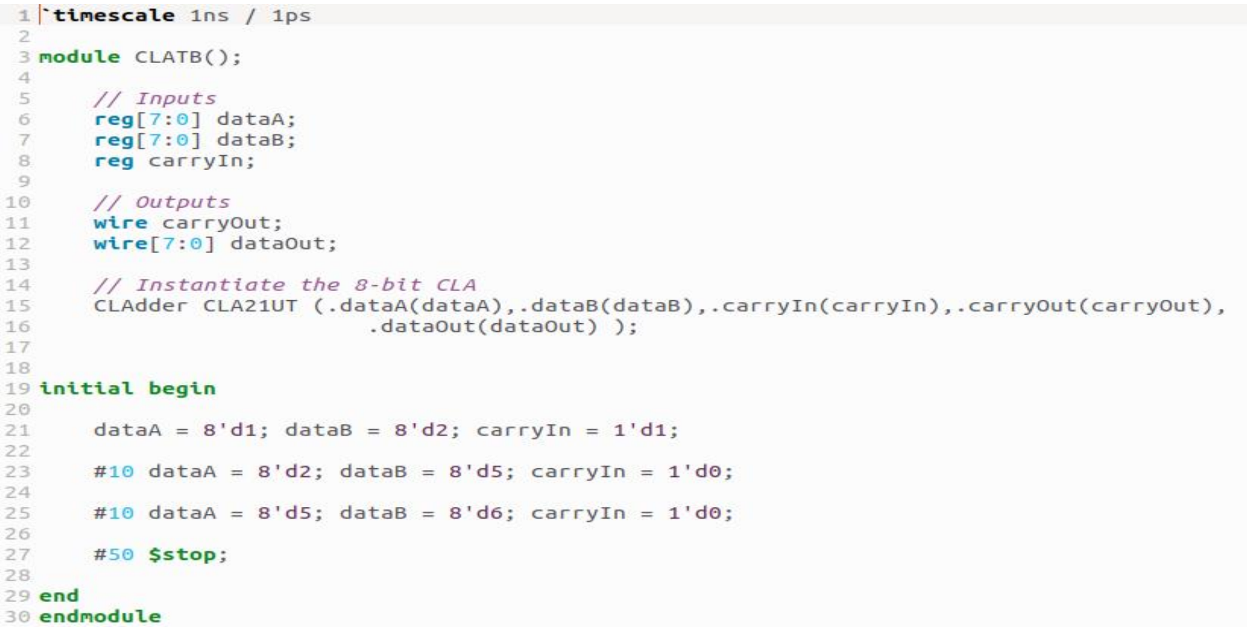
Marsa Shams 810198545

1-A,B)



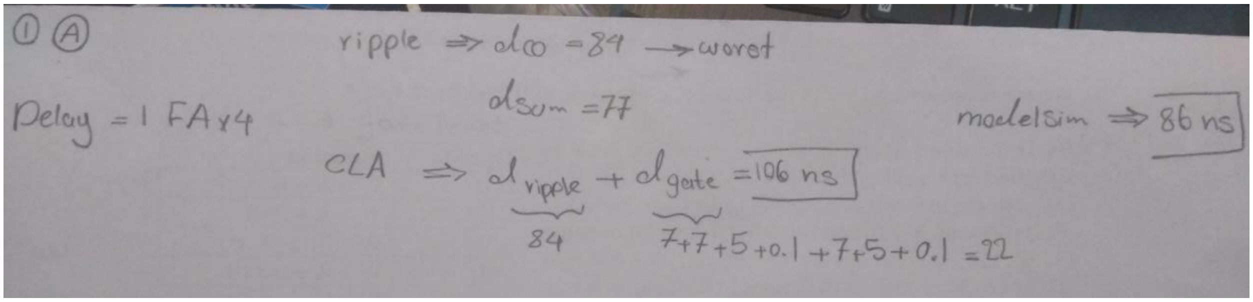
/home/marsa/Verilog/CA2/CA2-1/CA2-1.sv (/CLATB/CLA21UT) - Default *
Ln#
506000 ps

```
`timescale 1ns/1ns
module CLAdder(input [7:0] dataA,dataB ,input carryIn ,output [7:0] dataOut, output carryOut);

assign #106 {carryOut,dataOut} = dataA ^ dataB ^ carryIn ;



endmodule
```



```
`timescale 1ns / 1ps

module CLATB();

    // Inputs
    reg[7:0] dataA;
    reg[7:0] dataB;
    reg carryIn;

    // Outputs
    wire carryOut;
    wire[7:0] dataOut;

    // Instantiate the 8-bit CLA
    CLAdder CLA21UT (.dataA(dataA),.dataB(dataB),.carryIn(carryIn),.carryOut(carryOut),
                      .dataOut(dataOut) );


initial begin

    dataA = 8'd1; dataB = 8'd2; carryIn = 1'd1;

    #10 dataA = 8'd2; dataB = 8'd5; carryIn = 1'd0;

    #10 dataA = 8'd5; dataB = 8'd6; carryIn = 1'd0;

    #50 $stop;

end
endmodule
```

Delay:



① Ⓐ

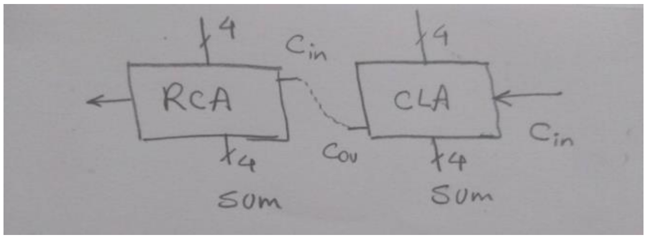
ripple ⇒ $d_{co} = 84$ → woret

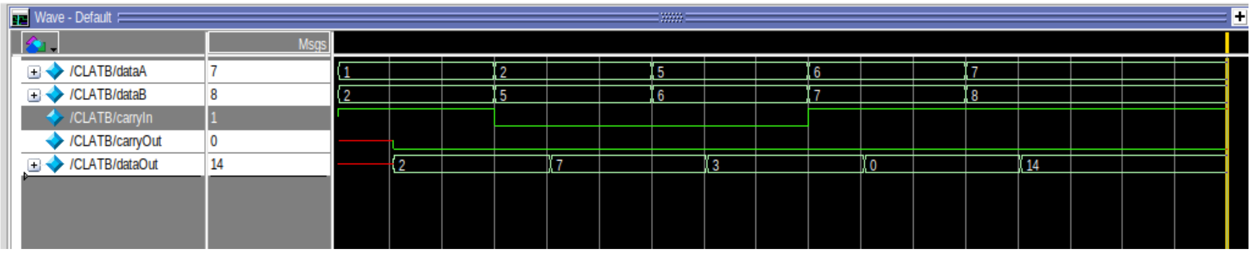
$d_{sum} = 77$

Delay = 1 FA × 4

modelsim ⇒ $\boxed{86\ ns}$

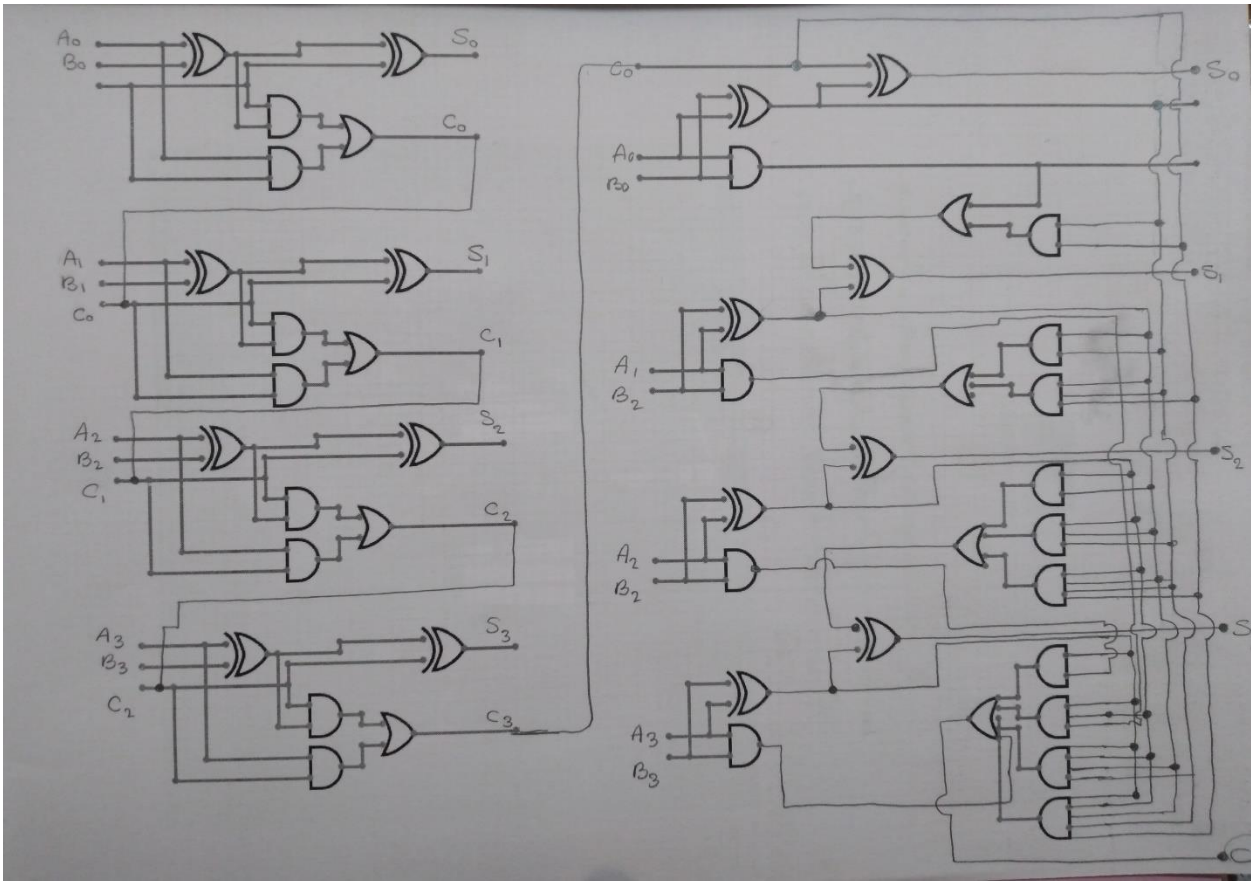
CLA ⇒ $\underbrace{d_{ripple}}_{84} + \underbrace{d_{gate}}_{7+7+5+0.1+7+5+0.1=22} = \boxed{106\ ns}$



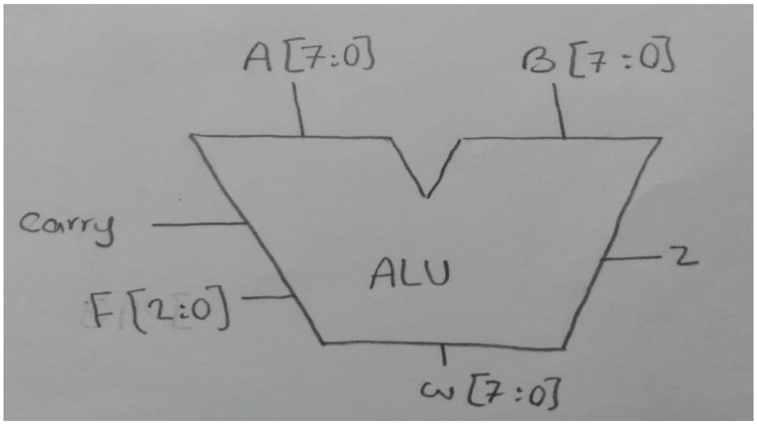


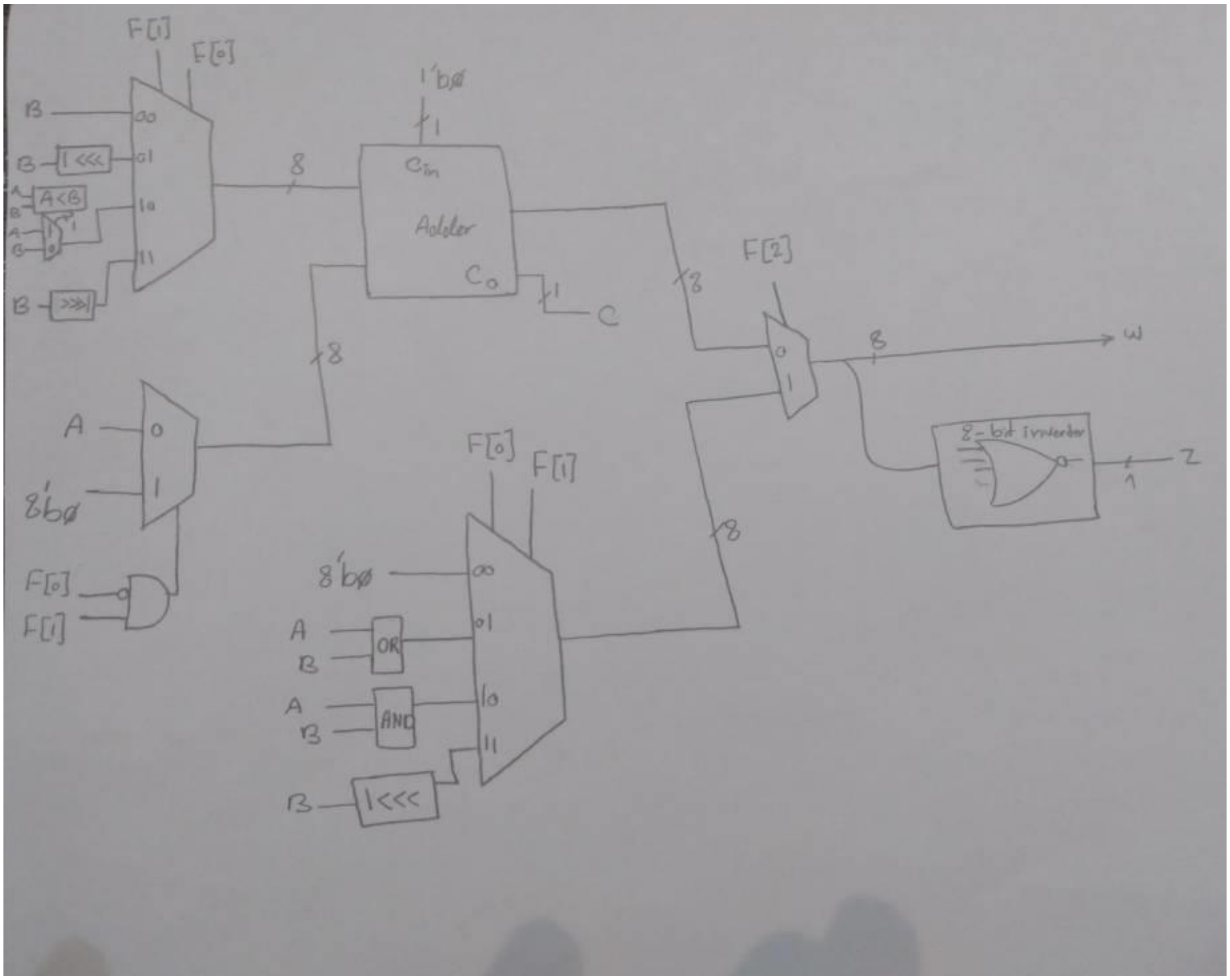
Wave - Default

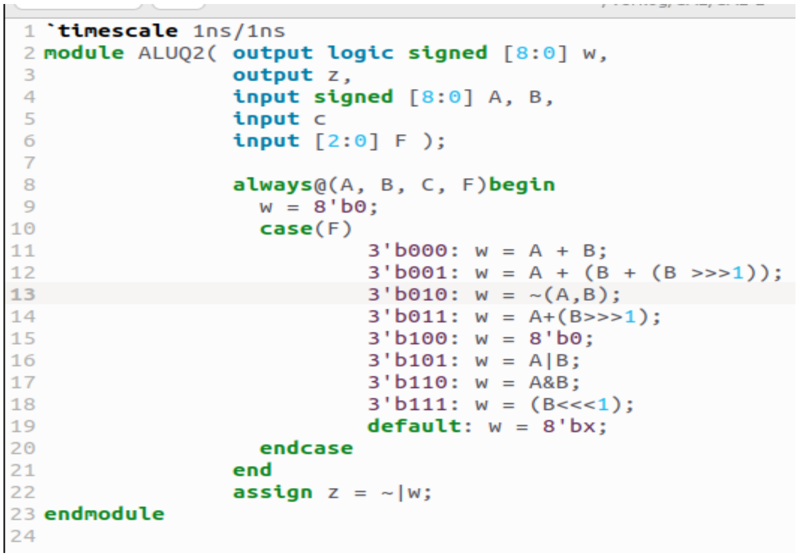
	Msgs
/CLATB/dataA	7
/CLATB/dataB	8
/CLATB/carryIn	1
/CLATB/carryOut	0
/CLATB/dataOut	14



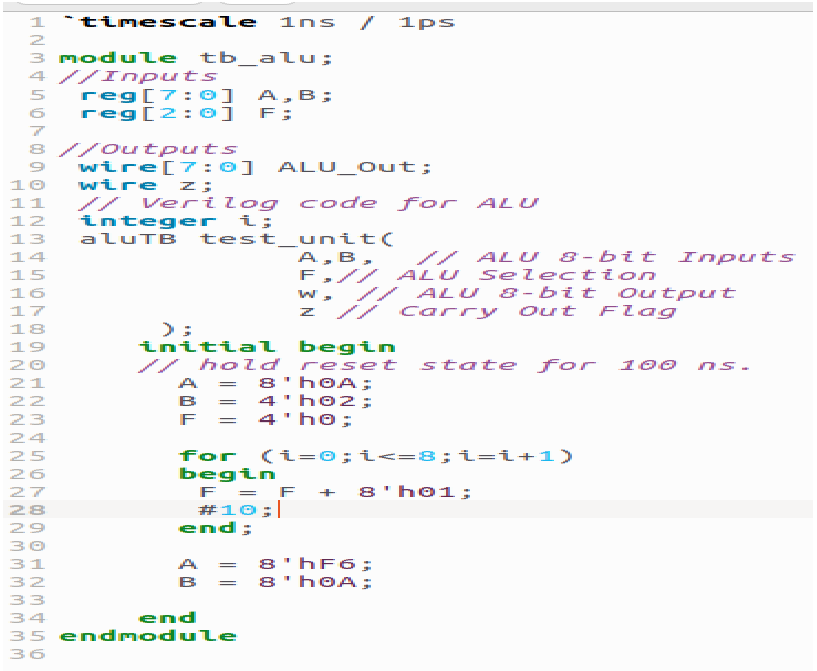
C,D,E)







```
`timescale 1ns/1ns
module ALUQ2( output logic signed [8:0] w,
              output z,
              input signed [8:0] A, B,
              input c
              input [2:0] F );

              always@(A, B, C, F)begin
                w = 8'b0;
                case(F)
                        3'b000: w = A + B;
                        3'b001: w = A + (B + (B >>>1));
                        3'b010: w = ~(A,B);
                        3'b011: w = A+(B>>>1);
                        3'b100: w = 8'b0;
                        3'b101: w = A|B;
                        3'b110: w = A&B;
                        3'b111: w = (B<<<1);
                        default: w = 8'bx;
                endcase
              end
              assign z = ~|w;
endmodule
```



```
`timescale 1ns / 1ps

module tb_alu;
//Inputs
 reg[7:0] A,B;
 reg[2:0] F;

//Outputs
 wire[7:0] ALU_Out;
 wire z;
 // Verilog code for ALU
 integer i;
 aluTB test_unit(
            A,B,  // ALU 8-bit Inputs
            F,// ALU Selection
            w, // ALU 8-bit Output
            z // Carry Out Flag
     );
    initial begin
    // hold reset state for 100 ns.
      A = 8'h0A;
      B = 4'h02;
      F = 4'h0;

      for (i=0;i<=8;i=i+1)
      begin
       F = F + 8'h01;
       #10;
      end;

      A = 8'hF6;
      B = 8'h0A;

    end
endmodule

```