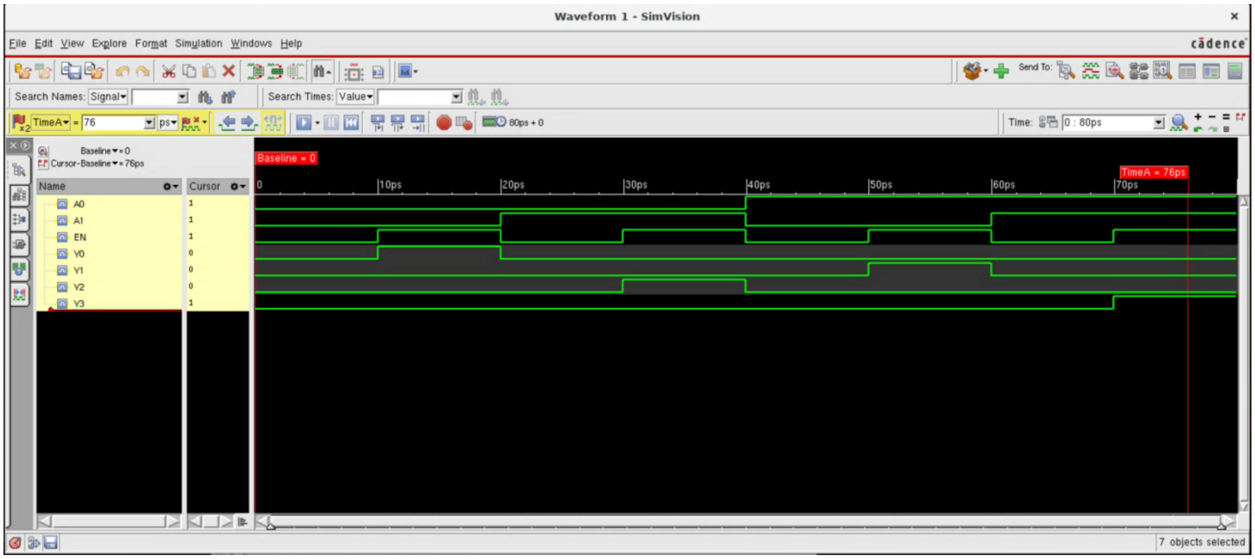
Problem 1:
Module:

```
`timescale 1ps/1ps

module p1(A0,A1,EN,Y0,Y1,Y2,Y3);
  input A0,A1,EN;
  output Y0,Y1,Y2,Y3;

  assign Y0 = ~A0&~A1&EN;
  assign Y1 = A0&~A1&EN;
  assign Y2 = ~A0&A1&EN;
  assign Y3 = A0&A1&EN;
endmodule
```

Test bench:

```
`timescale 1ps/1ps

module p1_tb;
  reg A0,A1,EN;
  wire Y0,Y1,Y2,Y3;

  p1 test(A0,A1,EN,Y0,Y1,Y2,Y3);

  always
  begin
    $dumpfile("p1.vcd");
    $dumpvars(0,p1_tb);

    A0 = 0;
    A1 = 0;
    EN = 0;
    #10
    A0 = 0;
    A1 = 0;
    EN = 1;
    #10
    A0 = 0;
    A1 = 1;
    EN = 0;
    #10
    A0 = 0;
    A1 = 1;
    EN = 1;
    #10
    A0 = 1;
    A1 = 0;
    EN = 0;
    #10
    A0 = 1;
    A1 = 0;
    EN = 1;
    #10
    A0 = 1;
    A1 = 1;
    EN = 0;
    #10
    A0 = 1;
    A1 = 1;
    EN = 1;
    #10
    $finish;
  end
endmodule
```



Name	Cursor
A0	1
A1	1
EN	1
Y0	0
Y1	0
Y2	0
Y3	1

Fig 1: Problem 1 showing all possible inputs/outputs

Problem 2:

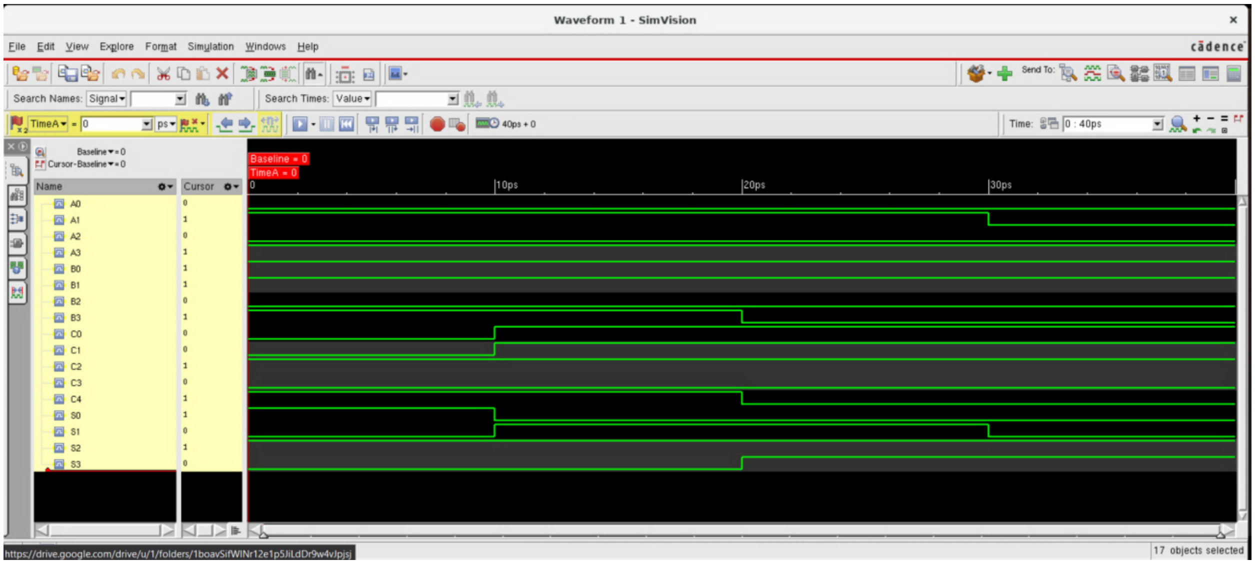
Module:

```
`timescale 1ps/1ps

module p2(A,B,CIN,S,COUT);
   input A,B,CIN;
   output S,COUT;

   assign S = (A^B)^CIN;
   assign COUT = (A&CIN)|(B&CIN)|(A&B);
endmodule
```

Test bench:

```
`timescale 1ps/1ps

module p2_tb;
   reg C0,A0,B0,A1,B1,A2,B2,A3,B3;
   wire S0,C1,S1,C2,S2,C3,S3,C4;

   p2 ADD0(A0,B0,C0,S0,C1);
   p2 ADD1(A1,B1,C1,S1,C2);
   p2 ADD2(A2,B2,C2,S2,C3);
   p2 ADD3(A3,B3,C3,S3,C4);

   always
   begin
      $dumpfile("p2.vcd");
      $dumpvars(0,p2_tb);

      A0 = 0;
      A1 = 1;
      A2 = 0;
      A3 = 1;

      B0 = 1;
      B1 = 1;
      B2 = 0;
      B3 = 1;

      C0 = 0;
      #10
      A0 = 0;
```

```
    A1 = 1;
    A2 = 0;
    A3 = 1;

    B0 = 1;
    B1 = 1;
    B2 = 0;
    B3 = 1;

    C0 = 1;
    #10
    A0 = 0;
    A1 = 1;
    A2 = 0;
    A3 = 1;

    B0 = 1;
    B1 = 1;
    B2 = 0;
    B3 = 0;

    C0 = 1;
    #10
    A0 = 0;
    A1 = 0;
    A2 = 0;
    A3 = 1;

    B0 = 1;
    B1 = 1;
    B2 = 0;
    B3 = 0;

    C0 = 1;
    #10
    $finish;
  end
endmodule
```



Name	Cursor
A0	0
A1	1
A2	0
A3	1
B0	1
B1	1
B2	0
B3	1
C0	0
C1	0
C2	1
C3	0
C4	1
S0	1
S1	0
S2	1
S3	0

Figure 2: Problem 2 showing 4 possible inputs/outputs

CE 303 HW#3

P3: a) To get from inputs A_0, B_0, C_0 to S_3 by the critical path we use 2 XOR in series per adder, meaning $2 \times 4 \times 3 = 24$ time units

To get from inputs A_0, B_0, C_0 to C_4 by the critical path we use 2 XOR per adder for the first 3 adders & then 1 AND & 1 OR in series, meaning $2 \times 3 \times 3 + 2 = 20$ time units

b) Given Sum delay is 2 times Carry delay:

↳ worst case goes through 20 FA, 13 carries & 7 sums

↳ $D_s = 2$ $D_c = 1$ → $13 + 7 \times 2 = 27$

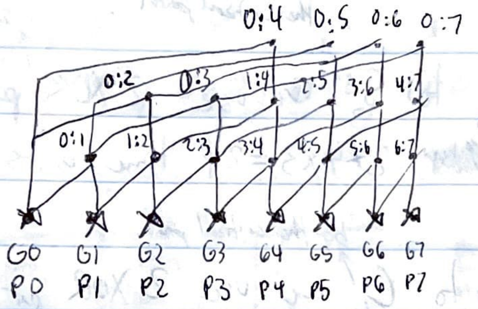
Given Carry delay is 2 times Sum delay:

↳ worst case goes through 20 FA, 14 carries & 6 sums

↳ $D_s = 1$ $D_c = 2$ → $14 \times 2 + 6 = 34$

↳ I would favor having faster carries

P4:



The critical path involves 3 dot products, and at each dot product the critical path goes through 1 AND gate and 1 OR gate and at the beginning G_0 & P_0 are created by a single gate each in parallel → 3×2+1 = 7 gates

↳ Critical Path:

$$\underset{\text{1st gate}}{A_0B_0} = G_0 \rightarrow \underset{\text{2nd gate}}{P_1G_0} \underset{\text{3rd gate}}{+G_1} = G_{0:1} \nearrow \underset{\text{4th gate}}{P_{2:3}G_{0:1}} \underset{\text{5th gate}}{+G_{2:3}} = G_{0:3}$$

↳ $G_{0:3} \rightarrow \underset{\text{6th gate}}{P_{4:7}G_{0:3}} \underset{\boxed{\text{7th gate}}}{+G_{4:7}} = G_{0:7}$

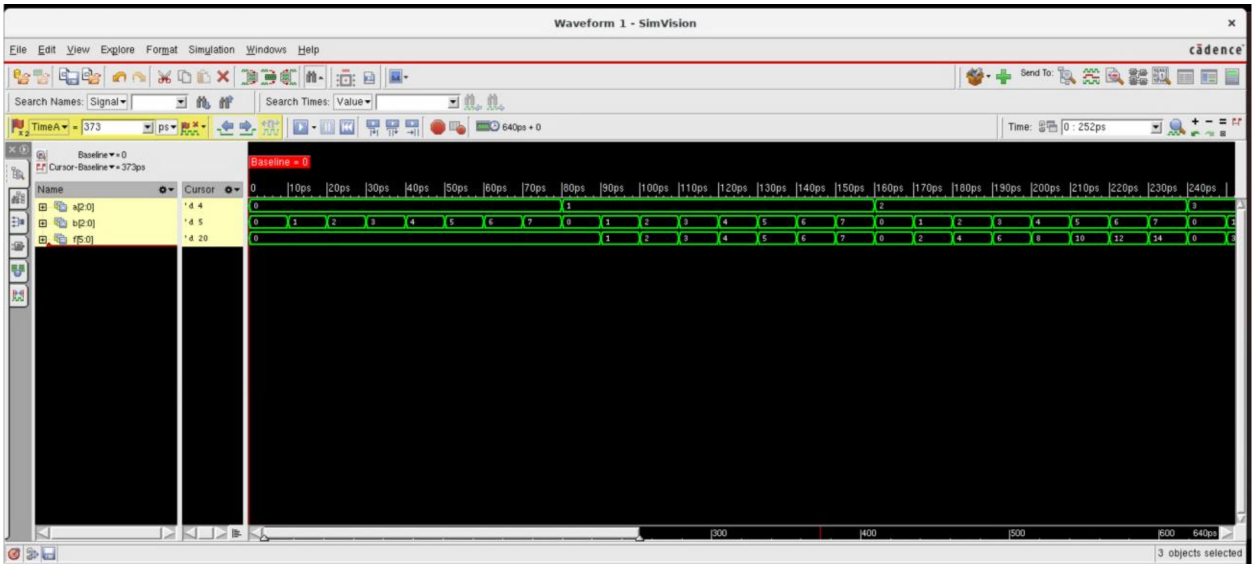
Problem 5
Module:

```
`timescale 1ps/1ps

module p5(
   input [2:0] a,
   input [2:0] b,
   output [5:0] f
);

   wire [2:0] s1;
   wire [2:0] s2;
   wire [3:0] p1;

   wire [2:0] s3;
   wire [3:0] p2;

   assign s1 = {a[2]&b[0], a[1]&b[0], a[0]&b[0]};
   assign s2 = {a[2]&b[1], a[1]&b[1], a[0]&b[1]};
   assign p1 = s2+s1[2:1];

   assign s3 = {a[2]&b[2], a[1]&b[2], a[0]&b[2]};
   assign p2 = s3+p1[3:1];

   assign f = {p2,p1[0],s1[0]};

endmodule
```

Test bench:

```
`timescale 1ps/1ps

module p5_tb;
   reg [2:0] a;
   reg [2:0] b;
   wire [5:0] f;

   p5 mult(a,b,f);

   initial begin
      $dumpfile("p5.vcd");
      $dumpvars(0,p5_tb);
```

```
    a = 0;
    b = 0;
    repeat(8) begin
      repeat(7) begin
        #10
        b = b+1;
      end
      #10
      a = a+1;
      b = 0;
    end
  end
endmodule
```



Waveform 1 - SimVision

Baseline = 0
Cursor-Baseline = 373ps

Name	Cursor
a[2:0]	'd 4
b[2:0]	'd 5
f[5:0]	'd 20

TimeA = 373 ps

Time: 0 : 252ps

Figure 3: Problem 5 view 1

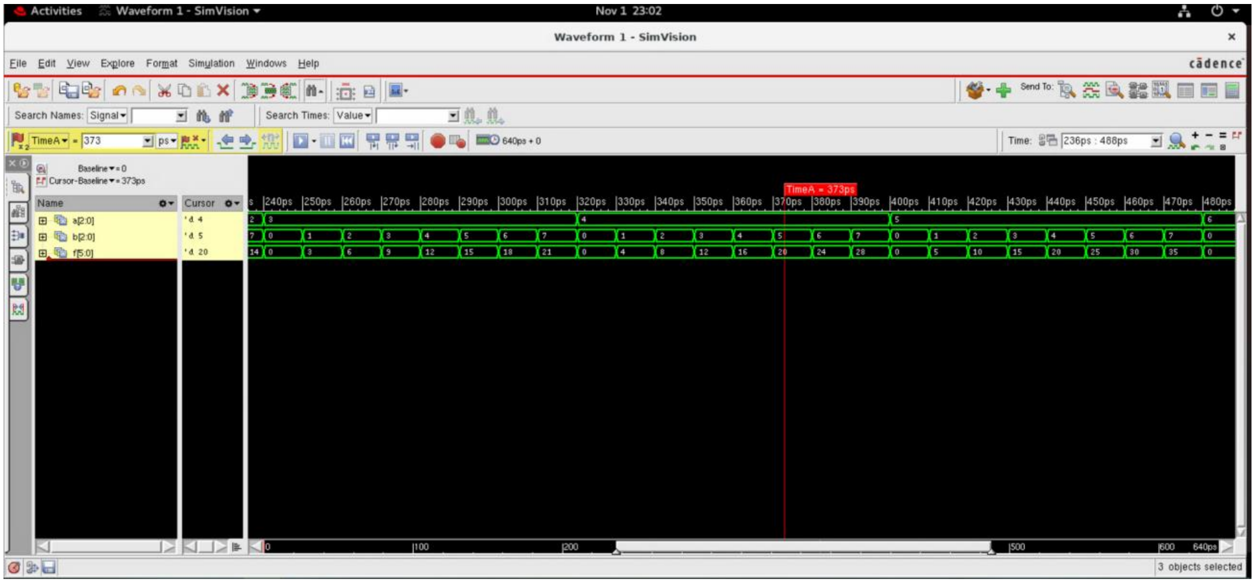


Figure 4: Problem 5 view 2

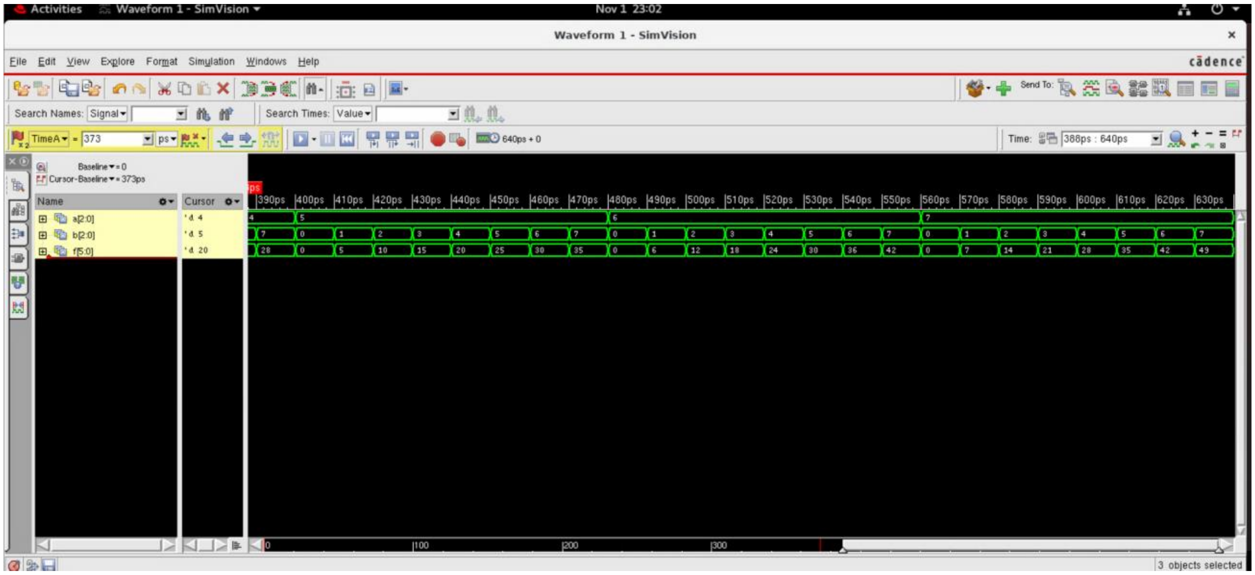


Figure 5: Problem 5 view 3