Assignment 1

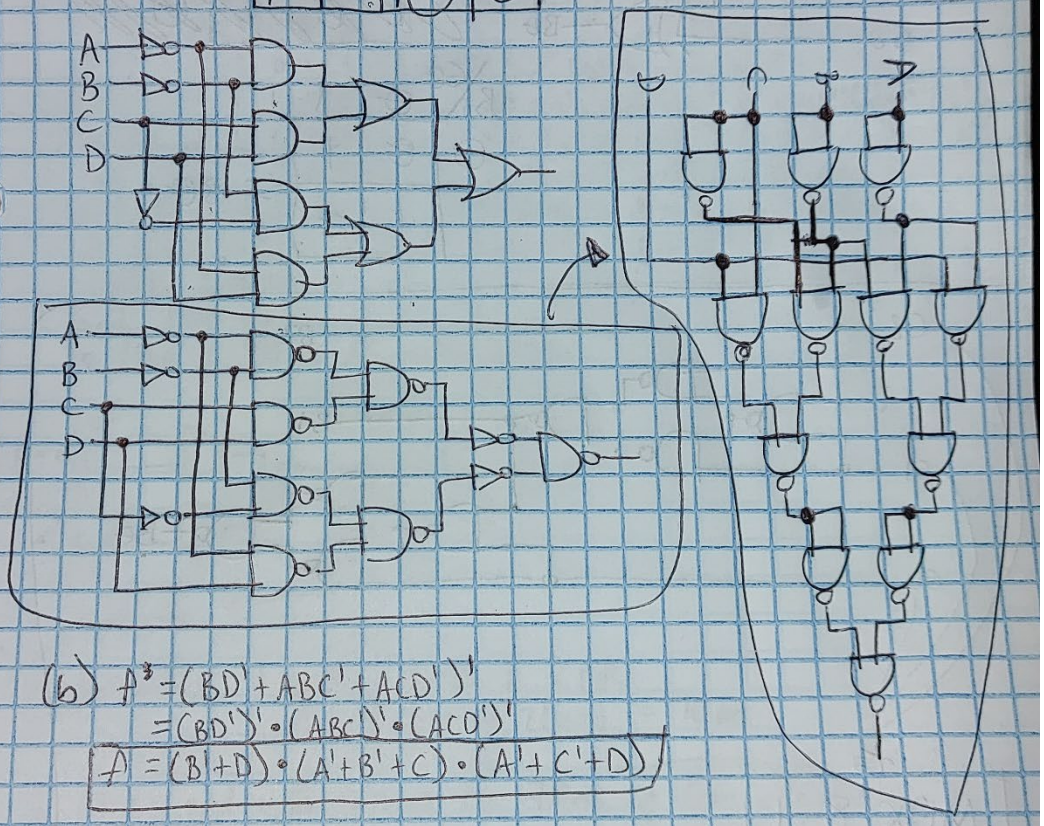
Oscar Osornio
EE 5193

1. (a)

AB \ CD	00	01	11	10
00	1	1	1	1
01	0	1	1	0
11	0	0	1	0
10	1	1	1	0

$f = A'B' + CD + B'C' + A'D$

$f' = BD' + ABC' + ACD'$

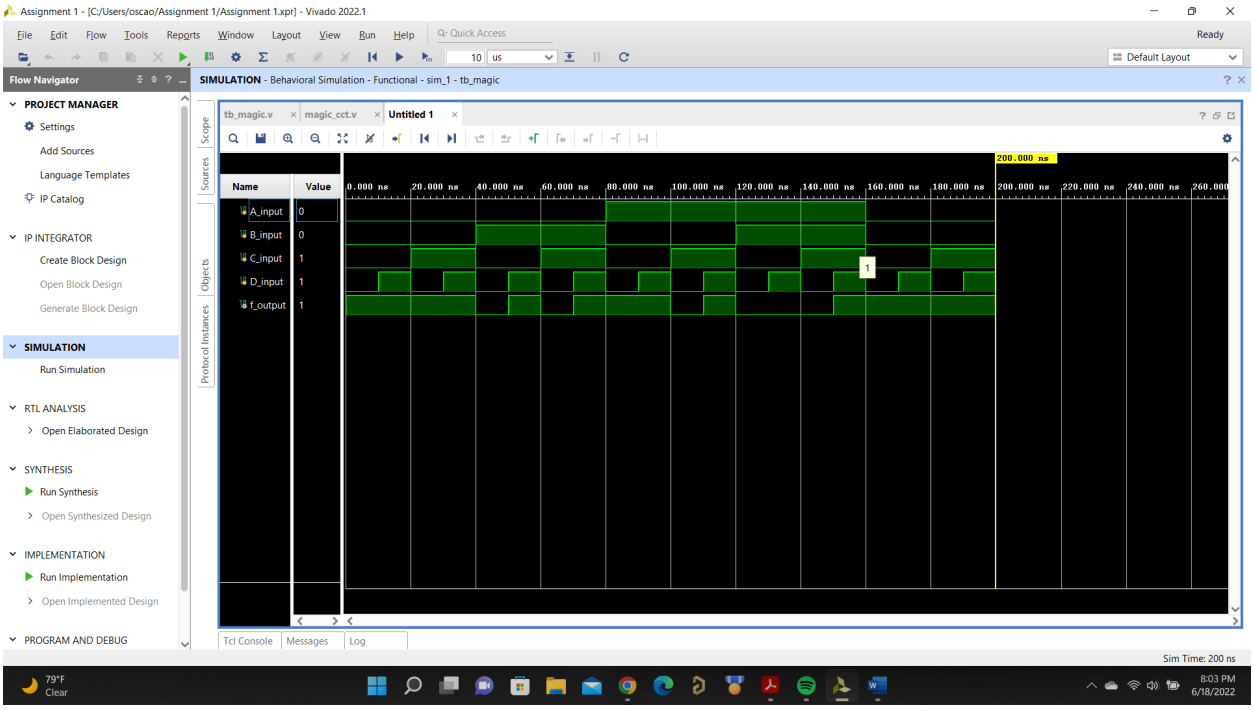


(b) $f' = (BD' + ABC' + ACD')'$

$= (BD')' \cdot (ABC')' \cdot (ACD')'$

$\boxed{f = (B + D) \cdot (A' + B' + C) \cdot (A' + C' + D)}$

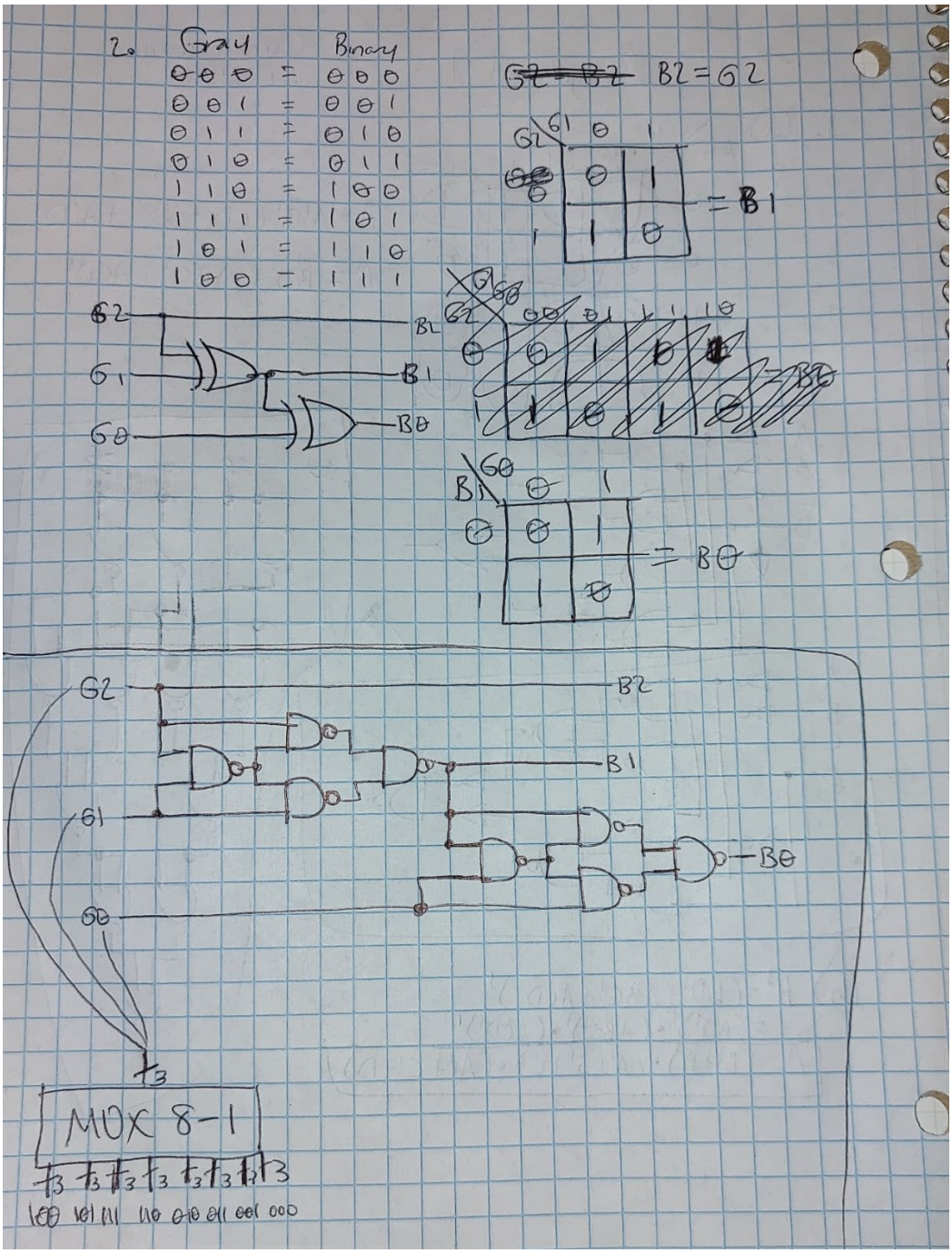
Problem 1(c)


Assignment 1 - [C:/Users/oscao/Assignment 1/Assignment 1.xpr] - Vivado 2022.1
File Edit Flow Tools Reports Window Layout View Run Help
Quick Access
Ready
Default Layout
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
Create Block Design
Open Block Design
Generate Block Design
SIMULATION
Run Simulation
RTL ANALYSIS
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
PROGRAM AND DEBUG
SIMULATION - Behavioral Simulation - Functional - sim_1 - tb_magic
tb_magic.v
magic_cct.v
Untitled 1
Name
Value
A_input
B_input
C_input
D_input
f_output
200.000 ns
Tcl Console
Messages
Log
Sim Time: 200 ns
79°F
Clear
8:03 PM
6/18/2022

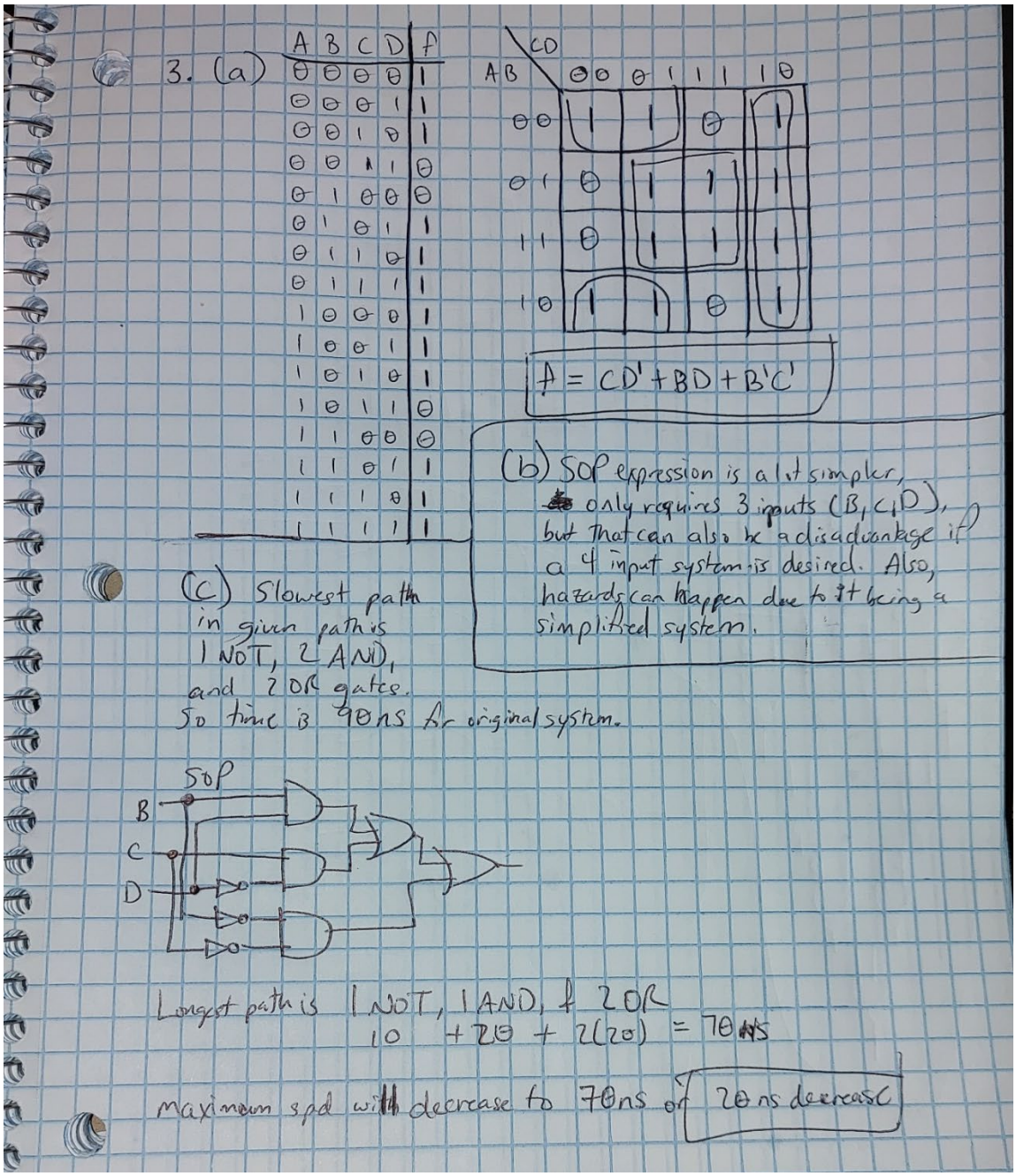
```
`timescale 1ns / 1ps

module magic_cct(
   input A,
   input B,
   input C,
   input D,
   output f
   );

   wire A, B, C, D;
   reg f;

   always @(A or B or C or D)
   begin
      if(A == 0 && B == 0)
      begin
         f = 1;
      end

      else if(C == 1 && D == 1)
      begin
         f = 1;
      end

      else if(B == 0 && C == 0)
      begin
         f = 1;
      end

      else if(A == 0 && D == 1)
      begin
         f = 1;
      end

      else
      begin
         f = 0;
      end
   end

endmodule
```

```
`timescale 1ns / 1ps

module tb_magic(
  );

  //declare testbench variables
  reg A_input, B_input, C_input, D_input;
  wire f_output;

  //instantiate the design module and connect to the testbench variables
  magic_cct instantiation(.A(A_input), .B(B_input), .C(C_input),
                  .D(D_input), .f(f_output));

  initial
    begin
      $dumpfile("xyz.vcd");
      $dumpvars;

      //set stimulus to test the code
      A_input = 0;
      B_input = 0;
      C_input = 0;
      D_input = 0;
      #200 $finish;
    end

  //provide the toggling input (truth table)
  //this acts as the clock input
  always #80 A_input = ~A_input;
  always #40 B_input = ~B_input;
  always #20 C_input = ~C_input;
  always #10 D_input = ~D_input;

  //display output if there's a change in input event
  always @(A_input or B_input or C_input or D_input)
    $monitor("At TIME(in ns) = %t, A = %d B = %d C = %d D = %d f = %d",
          $time, A_input, B_input, C_input, D_input, f_output);

Endmodule
```

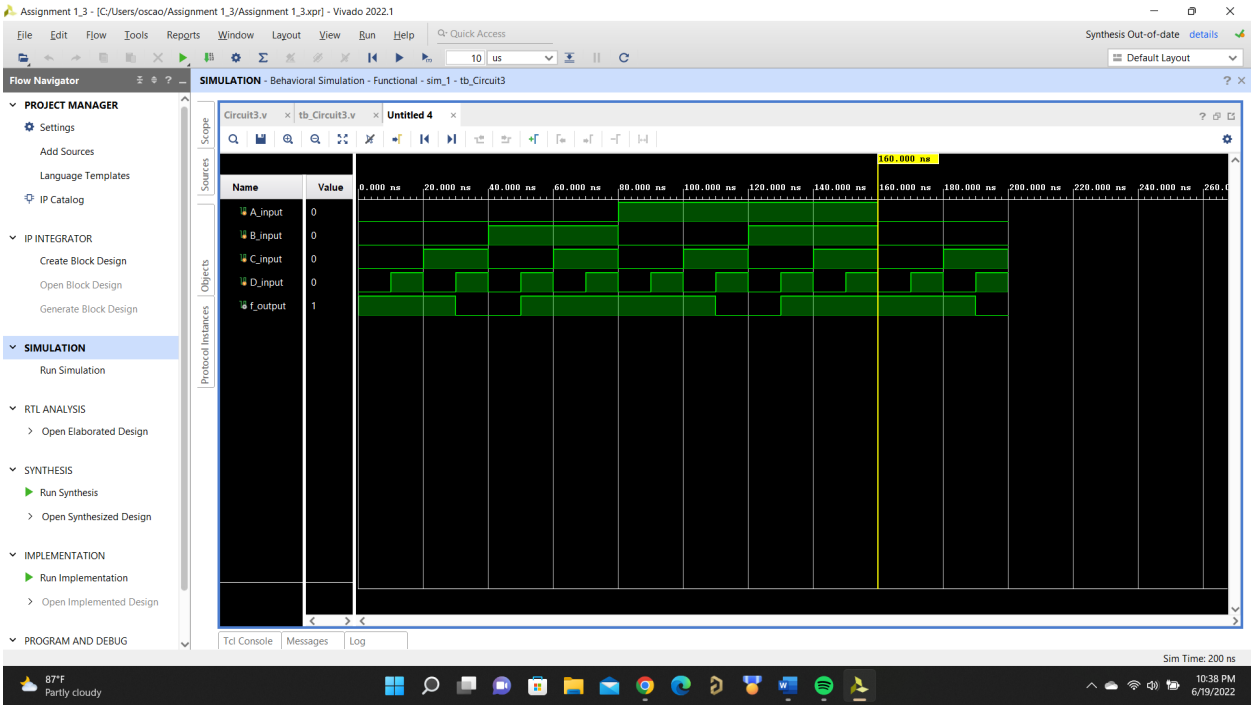
Problem 2



Problem 3


3. (a)
A B C D | f
CD
AB
f = CD' + BD + B'C'
(b) SOP expression is a lot simpler, only requires 3 inputs (B, C, D), but that can also be a disadvantage if a 4 input system is desired. Also, hazards can happen due to it being a simplified system.
(c) Slowest path in given path is 1 NOT, 2 AND, and 2 OR gates. So time is 90ns for original system.
SOP
B
C
D
Longest path is 1 NOT, 1 AND, & 2 OR
10 + 20 + 2(20) = 70ns
maximum spd with decrease to 70ns of 20ns decrease

Problem 3(d)


Assignment 1_3 - [C:/Users/oscao/Assignment 1_3/Assignment 1_3.xpr] - Vivado 2022.1
File Edit Flow Tools Reports Window Layout View Run Help
Quick Access
Synthesis Out-of-date details
Default Layout
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
Create Block Design
Open Block Design
Generate Block Design
SIMULATION
Run Simulation
RTL ANALYSIS
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
PROGRAM AND DEBUG
SIMULATION - Behavioral Simulation - Functional - sim_1 - tb_Circuit3
Circuit3.v
tb_Circuit3.v
Untitled 4
Name
Value
A_input
B_input
C_input
D_input
f_output
160.000 ns
Tcl Console
Messages
Log
Sim Time: 200 ns
87°F
Partly cloudy
10:38 PM
6/19/2022

```
`timescale 1ns / 1ps

module Circuit3(
   input wire A, B, C, D,
   output reg f
   );

   always @(A or B or C or D)
      begin

         if (C == 1 && D == 0)
            begin
               f = 1;
            end
         else if (B == 1 && D == 1)
            begin
               f = 1;
            end
         else if (B == 0 && C == 0)
            begin
               f = 1;
            end
         else
            begin
               f = 0;
            end

      end

endmodule
```

```
`timescale 1ns / 1ps

module tb_Circuit3(
  );

  //declare testbench variables
  reg A_input, B_input, C_input, D_input;
  wire f_output;

  //instantiate the design module and connect to the testbench variables
  Circuit3 instantiation(.A(A_input), .B(B_input), .C(C_input),
                  .D(D_input), .f(f_output));

  initial
    begin
      $dumpfile("xyz.vcd");
      $dumpvars;

      //set stimulus to test the code
      A_input = 0;
      B_input = 0;
      C_input = 0;
      D_input = 0;
      #200 $finish;
    end

  //provide the toggling input (truth table)
  //this acts as the clock input
  always #80 A_input = ~A_input;
  always #40 B_input = ~B_input;
  always #20 C_input = ~C_input;
  always #10 D_input = ~D_input;

  //display output if there's a change in input event
  always @(A_input or B_input or C_input or D_input)
    $monitor("At TIME(in ns) = %t, A = %d B = %d C = %d D = %d f = %d",
          $time, A_input, B_input, C_input, D_input, f_output);


endmodule
```