CSM 61C Spring 2020

Boolean Algebra

Exam Question Compilation Solutions

This document is a PDF version of old exam questions by topic, ordered from least difficulty to greatest difficulty.

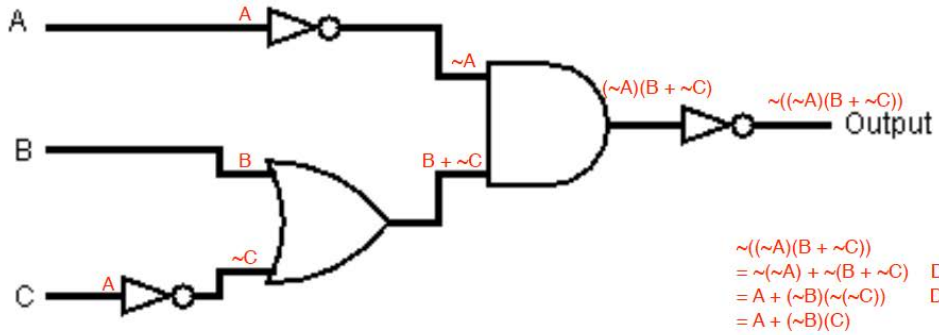
Questions:

- Fall 2015 Final QMT2-1a
- Fall 2018 Final QF2a
- Spring 2018 Midterm 2 Q2a
- Fall 2019 Midterm Q5b
- Fall 2019 Final Q5d
- Summer 2019 Midterm 2 Q3
- Summer 2018 Midterm 2 Q2

SID: _____

MT2-1: Synchronous Finite State Digital Machine Systems (9 points)

a. The circuit shown below can be simplified. Write a Boolean expression that represents the function of the simplified circuit using the minimum number of AND, OR, and NOT gates.



~((~A)(B + ~C))
= ~(~A) + ~(B + ~C) DeMorgan's Law
= A + (~B)(~(~C)) DeMorgan's Law
= A + (~B)(C)

~(~(A) (B+~(C))) = A + ~(B+(~C)) = A + (~B)C

F2) *SDS* (20 points = 8+5+7, 30 minutes)

a) Transform the **`fun`** function below into the *fewest Boolean gates* that implement the same logic. You may use AND, OR, XOR and NOT gates. *Hint: start with the truth table.*

```
bool fun(bool A, bool B) { return (A == B) ? true : B; }
```

a) If you write out the truth table, it's

A	B	fun(A,B)
FALSE	FALSE	TRUE
FALSE	TRUE	TRUE
TRUE	FALSE	FALSE
TRUE	TRUE	TRUE

...and using sum of products is:

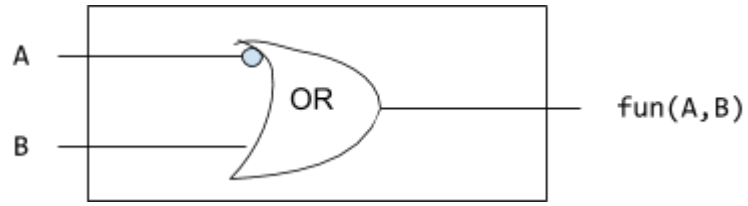
!A!B + !AB + AB

!A (!B + B) + AB distribution

!A + AB complimentarity, identity

!A + B [uniting theorem v.2: x + !xy = x + y (here x = !A, B = y)]

(the alternate exam swapped A and B. so they would have **A+!B** (the bubble would be on B, not A below).



Problem 2 ***Tell Us The Truth*** **(17 points)**

X	Y	Z	Out
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

(a) Select all of the following expressions that are equivalent to the truth table above.

- ● $(X + \bar{Y} + Z)(\bar{X} + \bar{Y} + Z)$
- ○ $\bar{Z} + Y$
- ● $\bar{Y} + Z$
- ○ $X\bar{Y}Z + \bar{X}\bar{Y}Z$
- ● $X\bar{Y} + \bar{X}\bar{Y} + Z\bar{X} + ZX$
- ● $\bar{Y} + \bar{Y}Z + Z$

Solution: Looking at the truth table, we see that there are more 0's in our output than 1's. This means that Product of Sums will be the easiest way to proceed.

Taking the Product of Sums will give us $(X + \bar{Y} + Z)(\bar{X} + \bar{Y} + Z)$. This is the first bubble.

We will then simplify this further to be: $(X\bar{X} + X\bar{Y} + XZ + \bar{Y}\bar{X} + \bar{Y}\bar{Y} + \bar{Y}Z + Z\bar{X} + \bar{Z}\bar{Y} + ZZ)$. We can use the boolean identities to simplify this to be $(\bar{Y}(X + \bar{X}) + Z(X + \bar{X}) + \bar{Y} + Z)$.

Since $(\bar{Y}(X + \bar{X})) = \bar{Y}$ and $(Z(X + \bar{X}) + \bar{Y} + Z) = Z)$, we can simplify this to be $(\bar{Y} + Z)$, which is option 3. This is the most simplified expression for the truth table.

We can also expand $(\bar{Y}(X + \bar{X})) + Z(X + \bar{X}))$ to be $(X\bar{Y} + \bar{X}\bar{Y} + Z\bar{X} + ZX)$, which is option 5.

For option 6, we see we can simplify by adding another $(\bar{Y}Z)$:

$(\bar{Y} + \bar{Y}Z + Z + \bar{Y}Z) = (\bar{Y}(Z + 1) + Z(\bar{Y} + 1) = \bar{Y} + Z$, which matches our simplified expression.

Option 2 is a simplified expression which does not match $\bar{Y} + Z$, so it must be incorrect. Option 4 is also incorrect, as it takes the Sum of Products approach on the 0 entries in our truth table.

This leaves options 1, 3, 5, and 6 as the correct answers.

(b) Suppose you wanted to implement $\bar{A} + B$, but the only available gates are NAND gates. What is the minimum number of NAND gates you need to implement the above truth table correctly?

Solution: 2

The boolean equation for A NAND B is $\bar{A} + \bar{B}$. This almost matches our expression $\bar{A} + B$, so we want to do A NAND $\bar{B}$.

If we try doing B NAND B, we see that this gives us $\bar{B}$. So, we can recreate the expression $\bar{A} + B$ by doing A NAND (B NAND B), which uses 2 NAND gates.

SOLUTIONS Fall 2019 Midterm

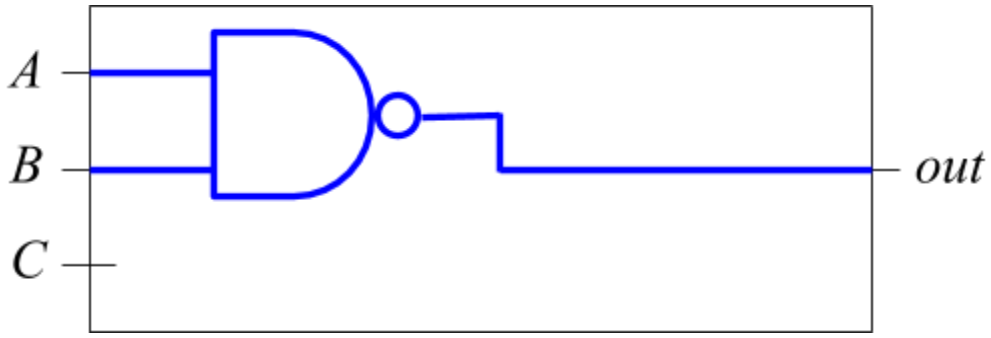
Question 5b

b) Draw the **FULLY SIMPLIFIED** (*fewest* number of primitive gates) circuit for the equation below. You may use the following primitive gates: AND, NAND, OR, NOR, XOR, XNOR, and NOT.

SHOW YOUR WORK FOR PART (b) BELOW

$$out = \overline{(A + \overline{B}B)} + \overline{(B + \overline{A})(A + BC)}$$

$out = \overline{(A)} + \overline{(B + \overline{A})(A + BC)}$	**Apply inverse law to $\overline{B}B$**
$out = \overline{(A)} + \overline{(BA + \overline{A}A + BBC + \overline{A}BC)}$	**Apply distributive law**
$out = \overline{(A)} + \overline{(BA + BC + \overline{A}BC)}$	**Apply inverse + idempotent laws**
$out = \overline{(A)} + \overline{(B(A + C + \overline{A}C))}$	**Apply distributive law**
$out = \overline{(A)} + \overline{(B)(A + C))}$	**Apply absorption law**
$out = \overline{(A)} + \overline{(B)} + \overline{(A + C)}$	**Apply DeMorgan's law**
$out = \overline{(A)} + \overline{(B)} + (\overline{A})(\overline{C})$	**Apply DeMorgan's law**
$out = \overline{(A)} + \overline{(B)}$	**Apply absorption law**
$out = \overline{(A)(B)}$	**Apply DeMorgan's law**



SOLUTIONS Fall 2019 Final
Question 5d

d) Draw the **FULLY SIMPLIFIED** (*fewest* primitive gates) circuit for the equation below into the diagram on the lower right. You may use the following primitive gates: AND, NAND, OR, NOR, XOR, XNOR, and NOT.

$$out = \overline{(C + AB\overline{C} + \overline{B}\,\overline{C}D)} + \overline{(C + \overline{B + D})}$$

SHOW YOUR WORK IN THIS BOX

out = $\overline{C}\ \overline{(AB\overline{C})}\ \overline{(\overline{B}\overline{C}D)} + \overline{C}(B + D)$ **(Demorgan's)**

out = $\overline{C}(\overline{A} + \overline{B} + C)(B + C + \overline{D}) + \overline{C}(B + D)$ **(Demorgan's)**

out = $(\overline{A}\,\overline{C} + \overline{B}\overline{C} + \overline{C}C)(B + C + \overline{D}) + B\overline{C} + \overline{C}D$ **(Distributive)**

out = $(\overline{A}\,\overline{C} + \overline{B}\overline{C})(B + C + \overline{D}) + B\overline{C} + \overline{C}D$ **(Inverse)**

out = $\overline{A}B\overline{C} + \overline{A}C\overline{C} + \overline{A}\,\overline{C}\,\overline{D} + B\overline{B}\overline{C} + BC\overline{C} + \overline{B}\,\overline{C}\,\overline{D} + B\overline{C} + \overline{C}D$ **(Distributive)**

out = $\overline{A}B\overline{C} + \overline{A}\,\overline{C}\,\overline{D} + \overline{B}\,\overline{C}\,\overline{D} + B\overline{C} + \overline{C}D$ **(Inverse)**

out = $B\overline{C}(A + 1) + \overline{A}\,\overline{C}\,\overline{D} + \overline{B}\,\overline{C}\,\overline{D} + \overline{C}D$ **(Distributive)**

out = $\overline{C}(B + \overline{A}\,\overline{D} + \overline{B}\overline{D} + D)$ **(Distributive)**

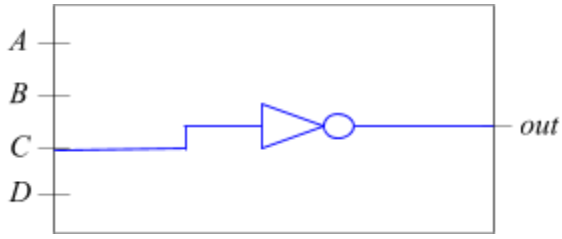
out = $\overline{C}((B + \overline{B}\,\overline{D}) + (\overline{A}\,\overline{D} + D))$ **(Associative)**

out = $\overline{C}(B + \overline{D} + \overline{A} + D)$

out = $\overline{C}(\overline{A} + B + (D + \overline{D}))$ **(Associative)**

out = $\overline{C}(\overline{A} + B + 1)$ **(Inverse)**

out = $\overline{C}$ **(Identity)**

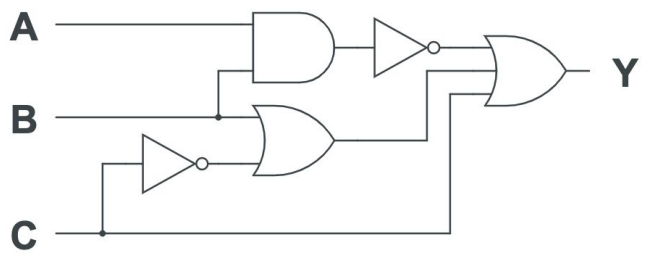


Question 3: Are Vulcans good at digital logic? - 14 pts

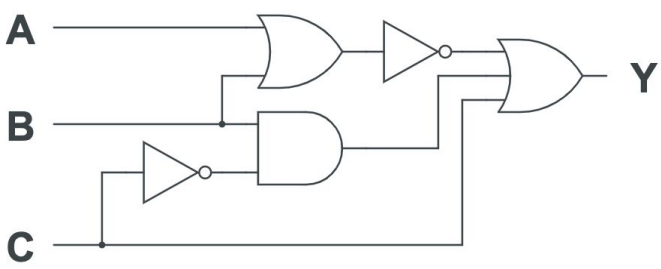
Which circuit diagram exactly matches the following boolean algebra expression?

$$Y = C(\overline{A+B})(B\overline{C})$$

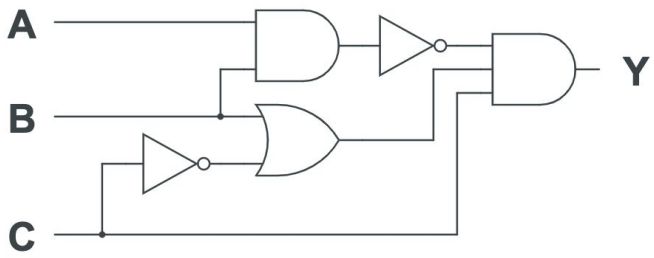
1)



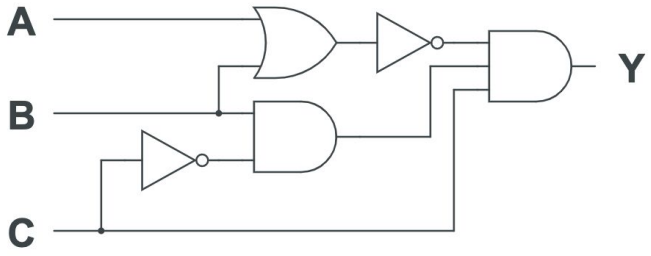
2)



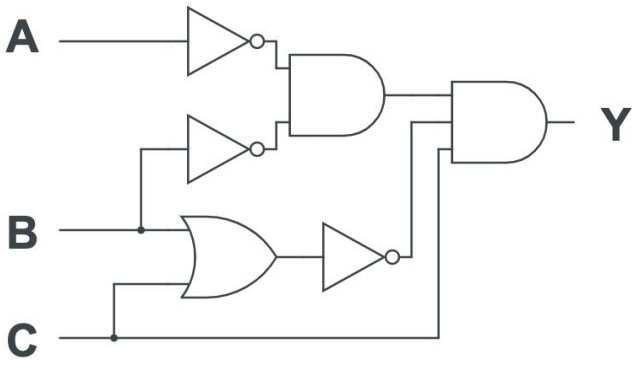
3)



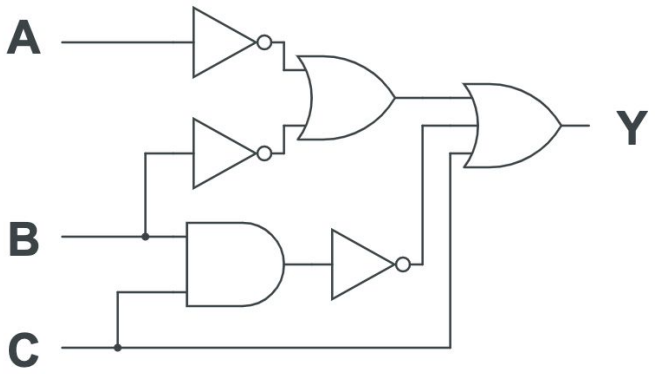
4)



5)



6)



The correct circuit is number: 4

Simplify the following boolean algebra expression. Show your work for partial credit, and you may use any method to simplify.

$$Y = B(AB + A\overline{B})(\overline{AC} + C)$$

$$Y = B(AB + A\neg B)(\neg(AC) + C)$$

$$Y = (ABB + A\neg BB)(\neg A + \neg C + C)$$

Because $XX = X$, $\neg XX = 0$, and $X + \neg X = 1$

$$Y = (AB + 0)(\neg A + 1)$$

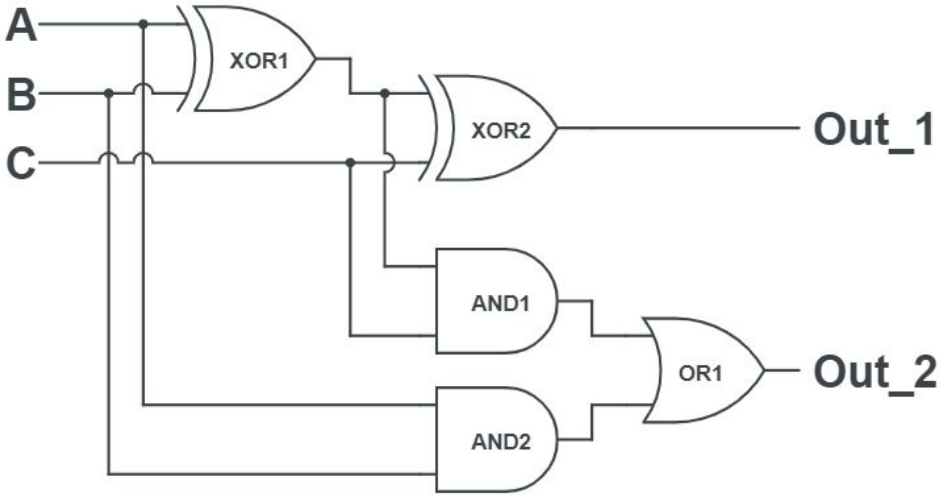
Because $X + 0 = X$, $\neg X + 1 = 1$

$$Y = AB(1)$$

Y=AB

Simplified Solution: AB

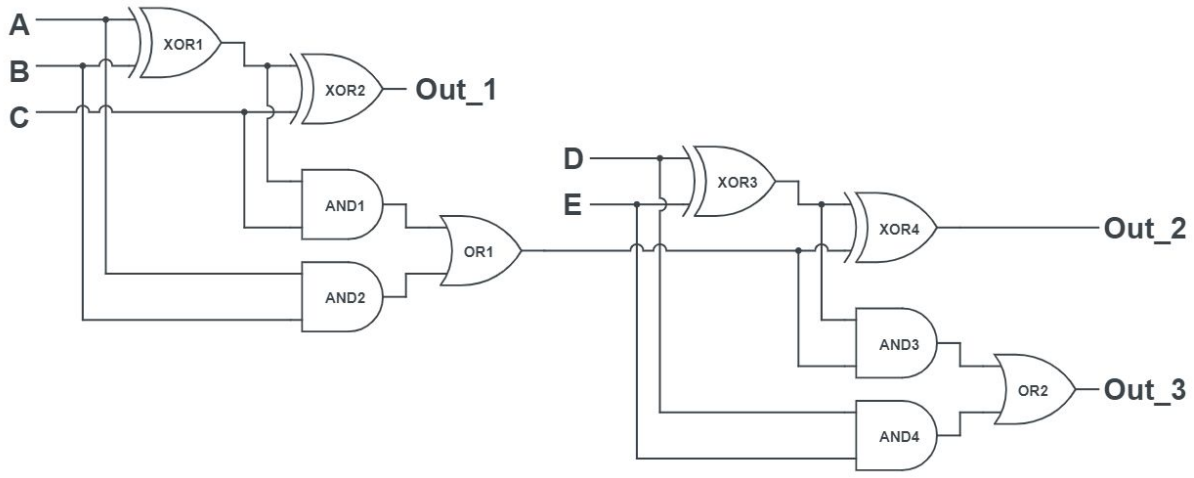
Fill out the following truth table that corresponds to the following circuit.



C	B	A	Out_1	Out_2
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Find the combination logic delays for each output or each circuit given the following parameters. There is no setup or hold time from the inputs or outputs.

- XOR gate delay: 80 ps
- AND gate delay: 60 ps
- OR gate delay: 40 ps



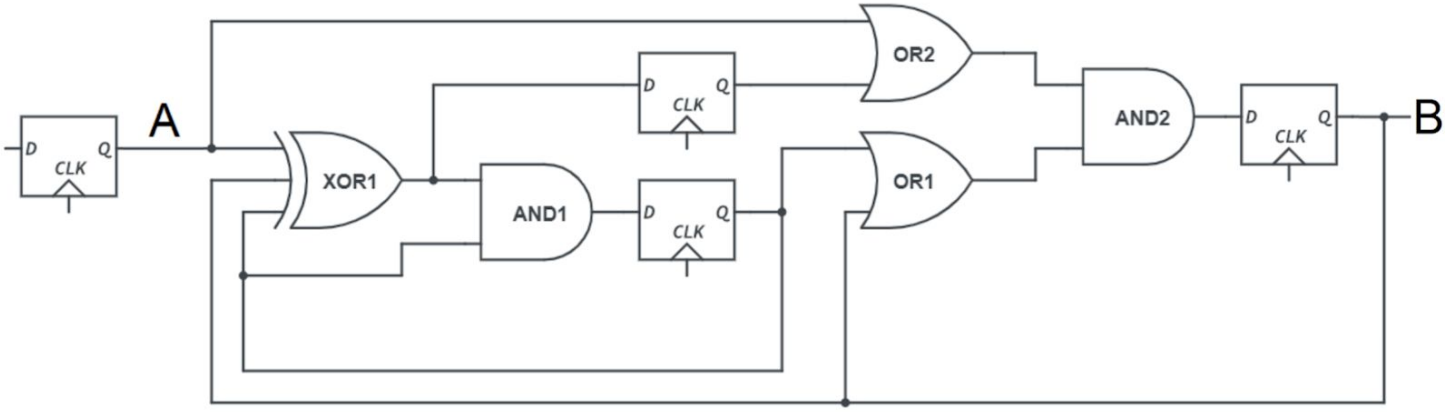
Out_1 Delay: 80ps + 80ps = 160ps

Out_2 Delay: 80ps + 60ps + 40ps + 80ps = 260ps

Out_3 Delay: 80ps + 60ps + 40ps + 60ps + 40ps = 280ps

For the next problems, consider the following pipelined circuit. Assume all registers have their clock inputs correctly connected to a global clock signal and that logic gates have the following parameters:

- XOR gate delay: 80 ps
- AND gate delay: 60 ps
- OR gate delay: 40 ps



When shopping for registers, we find two different models and want to determine which would be best for our circuit.

Register Type λ

- Setup Time: 40 ps
- Hold Time: 20 ps
- Clock-to-Q Delay: 30 ps

Register Type τ

- Setup Time: 10 ps
- Hold Time: 10 ps
- Clock-to-Q Delay: 80 ps

Critical Path = CLK_Q + XOR + AND + SETUP
Because this passes through 2 registers, our latency is 2 clock cycles.

Note after release we found 2 other interpretations to this question. 1 has just 1 critical path because it considers the latency to be just the top path A takes to B. The second also counts an extra clock to q to give A its value or propagate through the last register to B.

What is the minimum latency for the circuit from A to B if we use register type **λ**? 2 * (30ps + 80ps + 60ps + 40ps) = 420ps

What is the minimum latency for the circuit from A to B if we use register type **τ**? 2 * (80ps + 80ps + 60ps + 10ps) = 460ps

SID: _____

Question 2: Simple Democratic Selection (12 pts)

As the semester is reaching a close, Steven, Nick, and Damon are busy determining the difficulty of the final exam. All three will vote on whether the final should be easy or hard, but the final decision will always be made based on the following rules:

Rule 1. If the vote is unanimously hard or unanimously easy, then it will be hard or easy, respectively.
Rule 2. If Damon disagrees with Steven and Nick, then Damon's vote will be chosen.
Rule 3. If Steven and Nick differ, then the minority vote will be chosen.
Else In all other situations, the outcome can be either easy or hard (i.e. they can be anything)

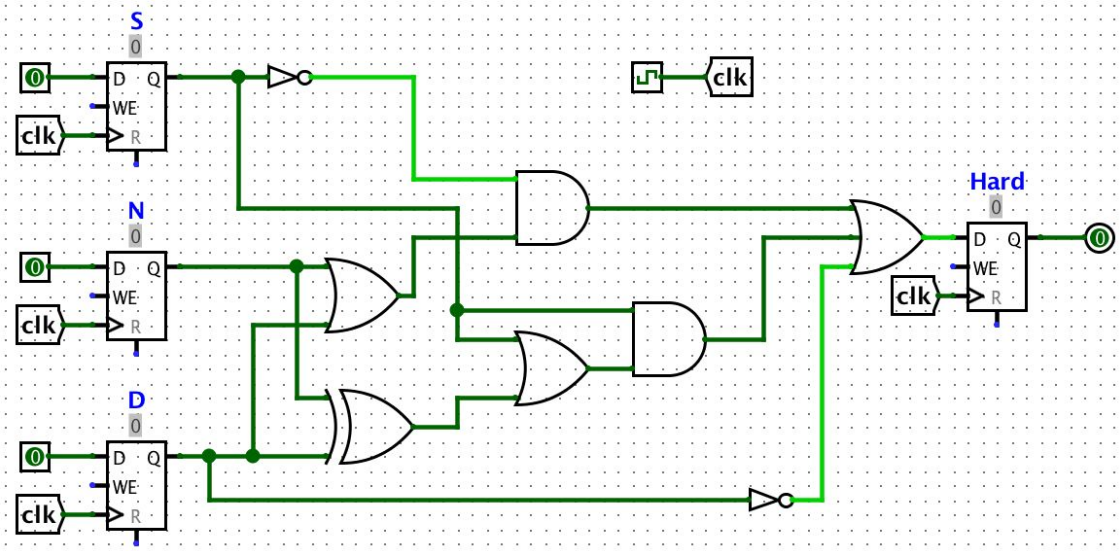
We will represent Steven's vote with the variable **S** which takes on values of 0 (easy) and 1 (hard). Similarly, Nick's vote is represented as **N** and Damon's vote is represented as **D**. For each rule, write out the **simplest** boolean logic expression using these three binary inputs that outputs whether or not the final exam will be easy or hard. Note: the symbol for XOR is $\oplus$.

Rule 1: SND **Rule2:** $(D \oplus S)\,(D \oplus N)\,D$ *or* $\overline{S \oplus N}\,D$ **Rule3:** $(S \oplus N)\,\overline{D}$

Below is a boolean algebra expression that models this problem. Simplify it into as few gates as possible:

$\overline{S}\,(N+D) + S(S + (N \oplus D)) + \overline{D}$ Simplified: $S(S + (N \oplus D) = S,\ \overline{S}\,(N+D) + S = S + N + D,\ 1$

For the next 2 questions, refer to the circuit diagram below which models the unsimplified boolean expression:



You are given that all logic gates have a propagation delay of 5ps, the register clk-to-q delay is 3ps, the register hold time is 11ps, and the setup time is 8ps. What is the critical path delay of this circuit?

Critical path delay: 3 + 5 + 5 + 5 + 5 + 8 = 31 ps

Using the assumptions above, what is the smallest value that the clk-to-q can be and **not** cause a hold-time violation?

clk-to-q: 11 - 5 + 5 = 1 ps