# **Section 4 Assignment (123 points) – Boolean Functions**

To receive credit, you must either show your work on the worksheet or explain how you got the answer.

80/123

1. (12 points) Indicate whether the following Boolean expressions are in conjunctive normal form (CNF), disjunctive normal form (DNF), both (B), or neither (N).
   1. (4 pts) CNF / DNF / B / N

CNF

* 1. (4 pts) CNF / DNF / B / N

B Here I think I just wrote down the wrong answer on accident

* 1. (4 pts) CNF / DNF / B / N

B

1. (10 points) Using the *Simplification Rules for Boolean Variables* and the *Laws of Boolean Algebra,* determine if the two Boolean expressions in each pair are equivalent:

~x+y+~x~y

~x+y

no they are not equivilant

1. (20 points) Give the Boolean function described by the following digital logic (circuit) diagrams
   1. (4 pts)



~(~x + ~y)

xy

* 1. (6 pts)





(x+y)+~(x(x+y)) here I used ~ instead of ‘ but it is the same answer

(x+y)+~(xx+xy)

(x+y)+~(x+xy)

x+y+~x

x+~x

true

* 1. (10 pts)



****

 ~((xy)+((x+y)(x+(yz)))) here I used ~ instead of ‘ but it is the same answer





1. (15 pts) For question 3c, use the laws of Boolean algebra to find a simpler circuit that computes the same function (you don’t need to draw it). How many gates would it use?

~((xy)+((x+y)(x+(yz))))

(~x~y)+~((x+y)(x+y)(x+z))

(~x~y)+~((x+y)(x+z))

(~x~y)+~(x+y)~(x+z)

(~x~y)+(~x~z)

it would require 7 gates

4 not

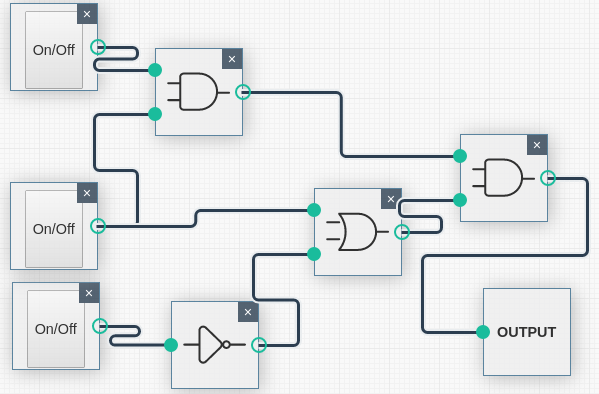
2 and

1 or

I guess I just didnt simplify it all the way or I didnt simplify it the right way. it does the same job though



1. (18 points) Draw the digital logic (circuit) diagrams for the following Boolean functions
   1. (6 pts)



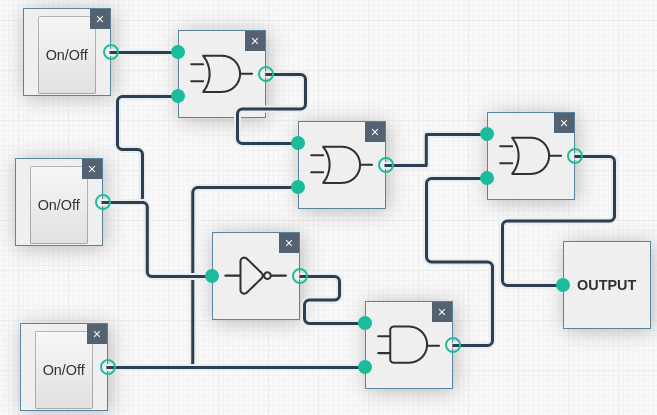
from top to bottom the inputs are:

x

y

z

* 1. (6 pts)



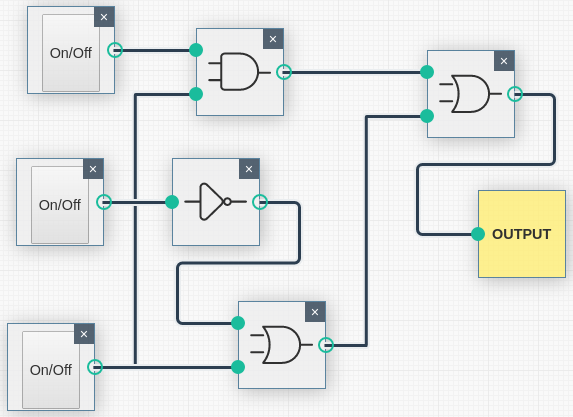
from top to bottom the inputs are:

x

y

z

* 1. (6 pts)



from top to bottom the inputs are:

x

y

z

I didnt realize that the bar above the y and z didnt appear correctly in the document: it showed up to me as xz+y’+z and not xz+(y+z)’ as it does in the pdf.

1. (18 points) For each Karnaugh map, use the *Kmap Grouping Rules* to group the elements and then list the simplified Boolean expression:
   1. (4 pts)

~x~y+xy



|  |  |  |
| --- | --- | --- |
|  | *y`* | *y* |
| x` | 1 |  |
| x |  | 1 |

* 1. (6 pts)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | *y`* | *y`* | *y* | *y* |
| x` |  |  |  |  |
| x | 1 | 1 | 1 | 1 |
|  | z` | z | z | z` |

x

* 1. (8 pts)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | *z`* | *z`* | *z* | *z* |  |
| x` | 1 | 1 | 1 | 1 | y` |
| x` | 1 | 1 | 1 | 1 | y |
| x |  |  |  | 1 | y |
| x | 1 |  |  | 1 | y` |
|  | w` | w | w | w` |  |

~x+~wz+~w~z~y

I didnt simplify it

1. (30 points) For each truth table, fill in the k-map, use the *Kmap Grouping Rules* to group the elements, and then list the simplified Boolean expression:

|  |  |  |
| --- | --- | --- |
| ***x*** | ***y*** | ***f(x,y)*** |
| 1 | 1 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 0 | 1 |

a. (4 pts)

|  |  |  |
| --- | --- | --- |
|  | *y* | *y`* |
| x |  | 1 |
| x` |  | 1 |

**~y**

b. (4 pts)

|  |  |  |
| --- | --- | --- |
| ***x*** | ***y*** | ***f(x,y)*** |
| 1 | 1 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |
| 0 | 0 | 1 |

|  |  |  |
| --- | --- | --- |
|  | *y* | *y`* |
| x | 1 |  |
| x` | 1 | 1 |

**y+~x**

****

|  |  |  |  |
| --- | --- | --- | --- |
| ***x*** | ***y*** | ***z*** | ***f(x,y,z)*** |
| 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 |

|  |  |  |
| --- | --- | --- |
|  | *z* | *z`* |
| xy | 1 | 1 |
| x`y | 1 | 1 |
| x`y` |  | 1 |
| xy` |  | 1 |

c. (6 pts) y+~z

d. (6 pts)

|  |  |  |  |
| --- | --- | --- | --- |
| ***x*** | ***y*** | ***z*** | ***f(x,y,z)*** |
| 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 |

|  |  |  |
| --- | --- | --- |
|  | *z* | *z`* |
| xy | 1 |  |
| x`y | 1 |  |
| x`y` | 1 |  |
| xy` | 1 |  |

x

e. (10 pts) zw(x+~y)+~xy(~z~w)+z~w(xy) Not simplified

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | *zw* | *z`w* | *z`w`* | *zw`* |
| xy | 1 | 0 | 0 | 1 |
| x`y | 0 | 1 | 1 | 1 |
| x`y` | 1 | 0 | 0 | 0 |
| xy` | 1 | 0 | 0 | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***x*** | ***y*** | ***z*** | ***w*** | ***f(x,y,z,w)*** |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

