Emma Gould
CS 270 - Circuits
Due: 2/4/16

Solution provided by Emma and Gabi

Homework #1

2.2

b) $(x + y)(x + y')$
$= xx + xy' + xy + yy' = x + x(y + y') + (0) = x + x(1) = x + x = x.$

c) $xyz + x'y + xyz'$
$= xy(z + z') + x'y = xy(1) + x'y = y(x + x') = y(1) = y.$

d) $(A + B)'(A' + B')'$
$= (A'B')(A''B'') = (A'B')(AB) = (A'A)(B'B) = 0.$

e) $(a + b + c')(a'b' + c)$
$= aa'b' + ac + ba'b' + bc + a'b'c' + c'c = (0) + ac + (0) + bc + a'b'c' + (0) = c(a+b) + a'b'c'.$

f) $a'bc + abc' + abc + a'bc'$
$= bc(a + a') + bc'(a + a') = bc(1) + bc'(1) = b(c + c') = b(1) = b.$

2.10

a) Show that $E = F_1 + F_2$ contains the sums of the minterms of F_1 and F_2.
We know that any algebraic expression is the sum of its minterms. Therefore,
$F_1 = \sum(m_{F1}), F_2 = \sum(m_{F2}), E = F_1 + F_2 = \sum(m_{F1}) + \sum(m_{F2}).$
Say, for example, $F_1 = \sum(m_{F1}) = m_1 + m_3 + m_4$ and $F_2 = \sum(m_{F2}) = m_2 + m_3 + m_5$.
This means $E = (m_1 + m_3 + m_4) + (m_2 + m_3 + m_5)$, and from the commutative property $((x + y) + z = x + (y + z))$, we know that:
$E = m_1 + m_3 + m_4 + m_2 + m_3 + m_5$, which is the sum of F_1 and F_2's minterms.

b) Show that $G = F_1F_2$ contains only the minterms common to F_1 and F_2.

Again, F_1 and F_2 are the sum of their minterms: $F_1 = \sum(m_{F1}), F_2 = \sum(m_{F2})$. Any two minterms will not be the same. Every minterm is a unique combination of the given variables, and as a result, ANDing two different minterms will always yield a zero. For example: $(abc)(a'bc)$, a cannot be both 0 and 1, and therefore this term is always zero. Another example: $(a'b'c')(abc')$, a and b cannot both be 0 and 1, so this term is also always zero. The only minterms that will yield a 1 are those that are the same. Example: $(ab'c)(ab'c)$, given that the terms are exactly the same, when a is 1, b is 0, and c is 1, both terms will be 1. ANDing the same term, according to Table 2.1: Postulates and Theorems of Boolean Algebra on page 43 of Digital Design, is the same as the term by itself. So,

$F_1 = \sum(m_{F1}), F_2 = \sum(m_{F2})$. Let's say $F_1 = m_1 + m_2 + m_4 + m_5$, and $F_2 = m_3 + m_4$, then
$G = F_1F_2 = (m_1 + m_2 + m_4 + m_5)(m_3 + m_4) = m_1m_3 + m_1m_4 + m_2m_3 + m_2m_4 + m_3m_4 + m_4m_4 + m_3m_5 + m_3m_4$
$G = m_4$ (the only minterm F1 and F2 share, because all other would always be zero)

2.11

a) $F = xy + xy' + y'z$. Simplified: $F = x(y + y') + y'z = x(1) + y'z = \underline{x + y'z}$.

Truth Table:

x	y	z	$y'z$	$x + y'z$
0	0	0	0	0
0	0	1	0	0
0	1	0	1	1
0	1	1	0	0
1	0	0	0	1
1	0	1	0	1
1	1	0	1	1
1	1	1	0	1

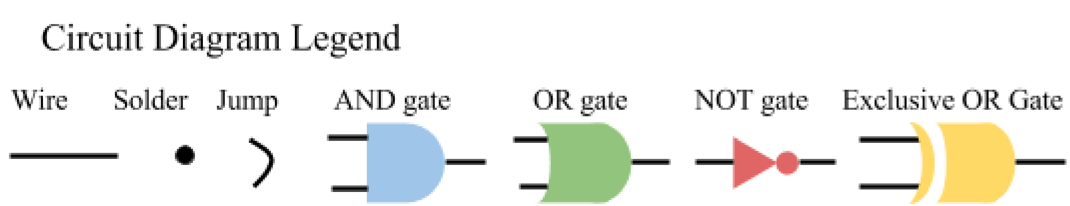
b) $F = bc + a'c'$

Truth Table:

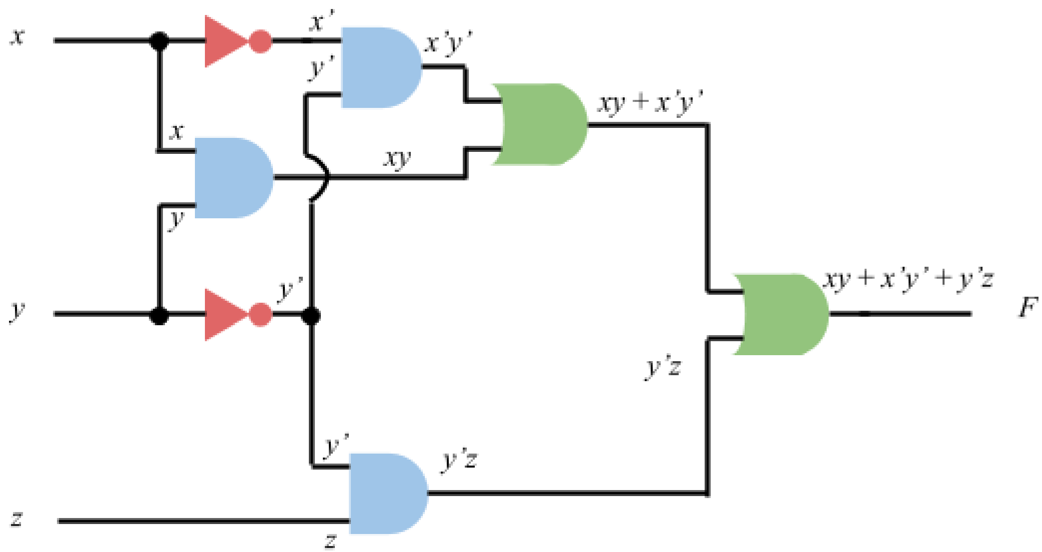
a	b	c	bc	$a'c'$	$bc + a'c'$
0	0	0	0	1	1
0	0	1	0	0	0
0	1	0	0	1	1
0	1	1	1	0	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	1	0	1

2.14

$F = xy + x'y' + y'z$



a) AND, OR, and inverter gates.



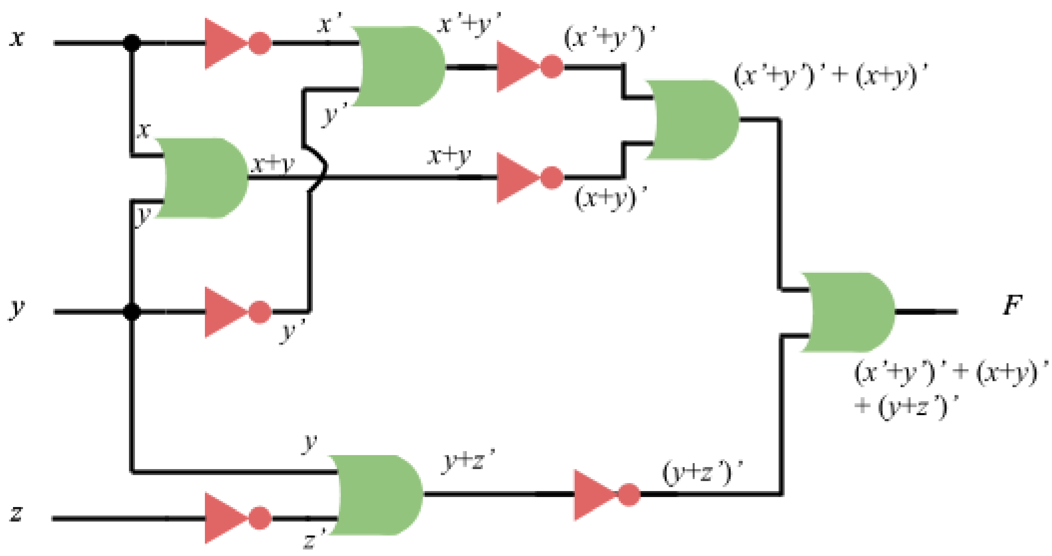
b) OR and inverter gates.

By DeMorgan's: $(ab)' = a' + b'$ => $(ab)'' = (a' + b')'$ => $ab = (a' + b')'$

$F = xy + x'y' + y'z = (x' + y')' + (x'' + y'')' + (y'' + z')' = (x' + y')' + (x + y)' + (y + z')'$

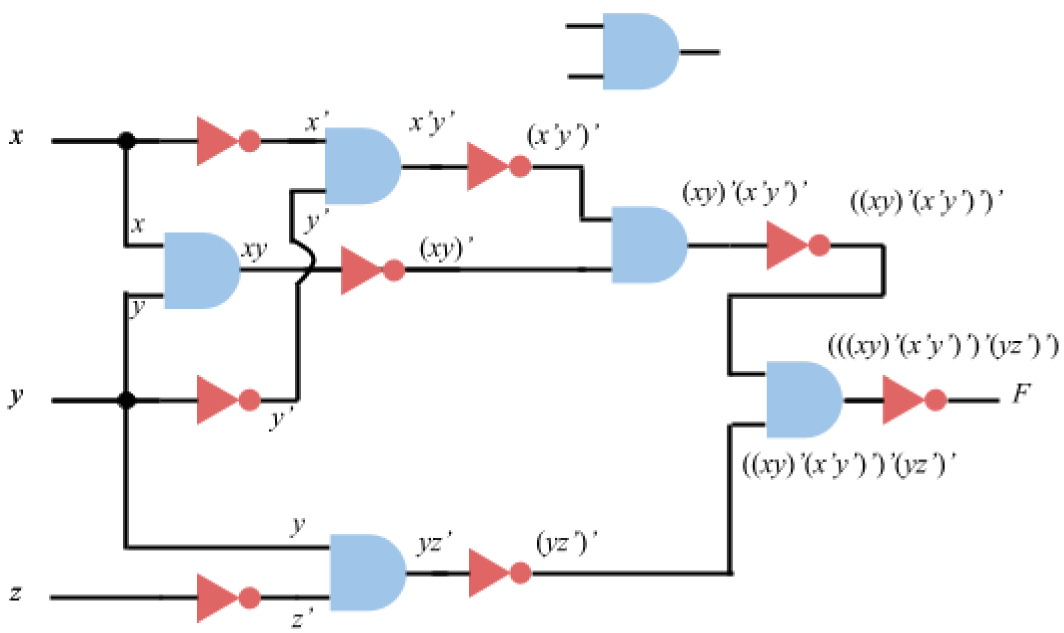
verify this expression is correct:

x	y	z	$xy + x'y' + y'z$	$x' + y'$	$x + y$	$y + z'$	$(x' + y')' + (x + y)' + (y + z')'$
0	0	0	1	1	0	1	1
0	0	1	1	1	0	0	1
0	1	0	0	1	1	1	0
0	1	1	0	1	1	1	0
1	0	0	0	1	1	1	0
1	0	1	1	1	1	0	1
1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1



c) AND and inverter gates

$F = xy + x'y' + y'z = ((xy)'(x'y')')' + y'z = ((((xy)'(x'y')')')'(y'z)')' = (((xy)'(x'y')')'(y'z)')'$



Python for Fig. 2.4

```
# G.E. Ranieri

# CSC270 HW1

# based on truthtable.py by D. Thiebaut

# generates a truth table for function f

# Let f = AB + C(D + E)


def f(a,b,c,d,e):

    return ( (a and b) or (c and (d or e)) )


def main():

    print( "  a  b  c  d  e  |  f  " )

    print( "----------------+-----" )

    for a in [0,1]:

        for b in [0,1]:

            for c in [0,1]:

                for d in [0,1]:

                    for e in [0,1]:

                        print( "%3d%3d%3d%3d%3d  |%3d" %

                              ( a, b, c, d, e, f(a,b,c,d,e) )

                               )

main()
```

Output:

```
   a  b  c  d  e  |  f
------------------+-----
   0  0  0  0  0  |  0
   0  0  0  0  1  |  0
   0  0  0  1  0  |  0
   0  0  0  1  1  |  0
   0  0  1  0  0  |  0
   0  0  1  0  1  |  1
   0  0  1  1  0  |  1
   0  0  1  1  1  |  1
   0  1  0  0  0  |  0
   0  1  0  0  1  |  0
   0  1  0  1  0  |  0
   0  1  0  1  1  |  0
   0  1  1  0  0  |  0
   0  1  1  0  1  |  1
   0  1  1  1  0  |  1
   0  1  1  1  1  |  1
   1  0  0  0  0  |  0
   1  0  0  0  1  |  0
   1  0  0  1  0  |  0
   1  0  0  1  1  |  0
   1  0  1  0  0  |  0
   1  0  1  0  1  |  1
   1  0  1  1  0  |  1
   1  0  1  1  1  |  1
   1  1  0  0  0  |  1
   1  1  0  0  1  |  1
   1  1  0  1  0  |  1
   1  1  0  1  1  |  1
   1  1  1  0  0  |  1
   1  1  1  0  1  |  1
   1  1  1  1  0  |  1
   1  1  1  1  1  |  1
```