***Problems based on simulations***

1. **For the function F(x, y, z) = xy +x’z + yz, identify the redundant term and then take 0,1,2 digits and convert them into equivalent letters (0 to 9 into A to J). Drive and design the new function with mapped letters highlighting the redundant term..ns if you want with justification.**

***Solution:***

Here, we have three variables x, y and z and all are repeated twice. The variable x is present in complemented form. So, all the conditions are satisfied for applying the Redundancy Theorem. According to redundancy theorem, the function

is equivalent to,

The yz term is redundant. Below is a proof for this,

(since (1 + A) = 1)

Hence, the redundant term is yz.

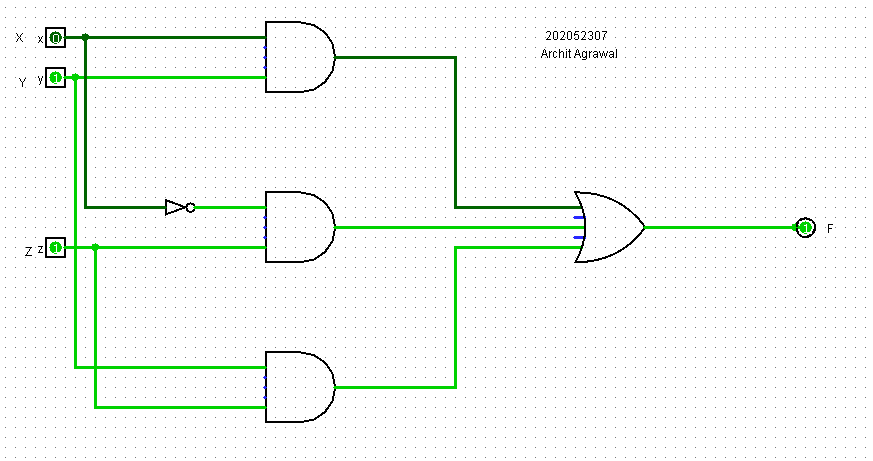
Let us take equivalent letters for 0, 1, 2 (0 to 9 into A to J). Therefore, 0 is A, 1 is B and 2 is C.

Therefore, the new expression is,

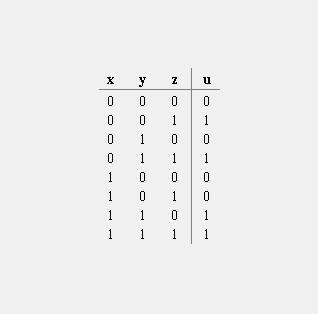
Now, let us verify these two Boolean Expressions using Logisim.

Firstly, the given Boolean expression is simulated in Logisim.

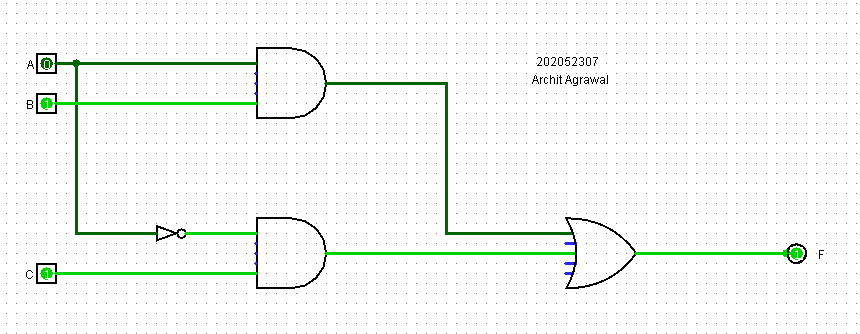
F = xy + x’z + yz



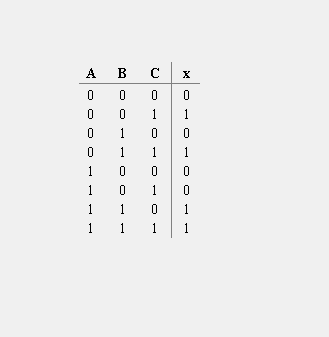
the truth table for the above expression is,



Now, let us design the circuit for the expression we formed after using redundancy theorem, i.e. F = AB + A’C



the truth table for the above circuit is,



As we can see that the truth table for both the expressions is same, hence, both the expressions are logically equivalent. Hence, we can conclude that the yz term in the given expression is redundant.

1. **Represent each of the following sentences by a Boolean equation.**

**(a) The company safe should be unlocked only when Mr. XYZ is in the office or Mr. ABC is in the office, and only when the company is open for business, and only when the security guard is present.**

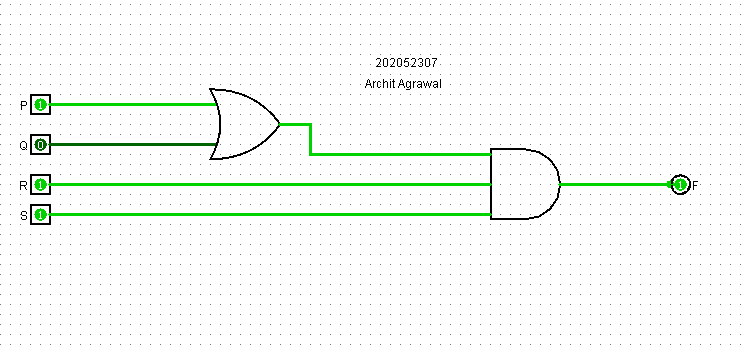
**(b) You should wear your overshoes if you are outside in a heavy rain and you are wearing your new suede shoes, or if your mother tells you to.**

***Solution:***

1. Let the below statements be represented by a logical variables as follows:

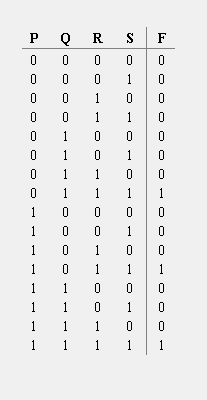
* Mr. XYZ is in the office: P
* Mr. ABC is in the office: Q
* The company is open for business: R
* The security guard is present: S
* The company safe should be unlocked: F

The Boolean Expression, therefore, will be:

Building a circuit for the above expression in Logisim.

This particular screenshot represents that Mr.XYZ is in the office, the company is open for business, and the security guard is present. Hence, the company’s safe is unlocked

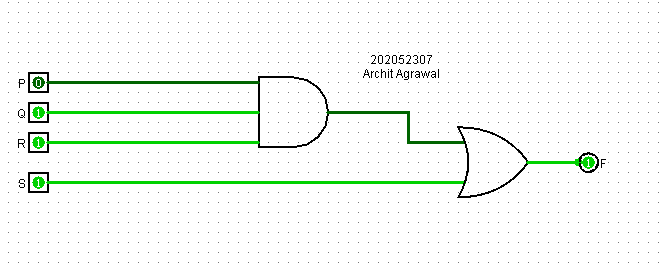
Analysing the circuit to get its truth table.



1. Let the below statements be represented by a logical variables as follows:

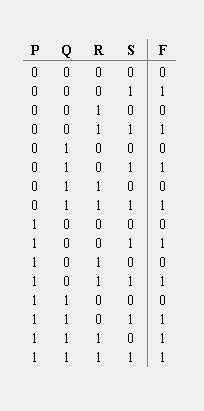
* You are outside heavy rain: P
* There is a heavy rain outside: Q
* You are wearing your new suede shoes: R
* Your mother tells you to wear them: S
* You should wear your overshoes: F

The Boolean Expression, therefore, will be:

Building a circuit for the above expression in Logisim.

This particular screenshot represents that there is heavy rain outside and you are wearing your new suede shoes, but your mother tells you to wear your overshoes. So, you are wearing your overshoes.

Analysing the circuit to get its truth table.



1. **Each of three coins has two sides, heads and tails. Represent the heads or tails status of each coin by a logical variable (A for the first coin, B for the second coin, and C for the third) where the logical variable is 1 for heads and 0 for tails. Write a logic function F(A, B, C) which is 1 if exactly one of the coins is heads after a toss of the coins. Express F**

**(a) as a minterm expansion.**

**(b) as a maxterm expansion.**

***Solution:***

The heads or tails status of the coin is represented by a logical variable:

* ‘A’ for the first coin
* ‘B’ for the second coin
* ‘C’ for the third coin

The head status is represented by 1 and the tails status by 0.

Since, F(A, B, C) is 1 if exactly one of the coins is heads after a toss of the coins, the truth table for F should look like

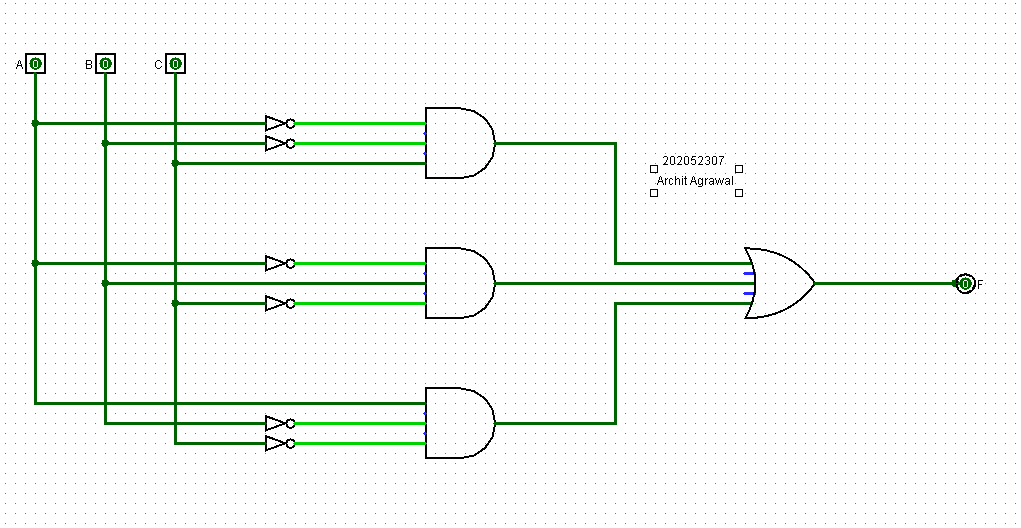
|  |  |  |  |
| --- | --- | --- | --- |
| A | B | C | F |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

In the output column,

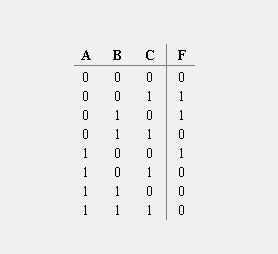
* first 1 entry is because only coin C is heads
* second 1 entry is because only coin B is heads
* third 1 entry is because only coin A is heads

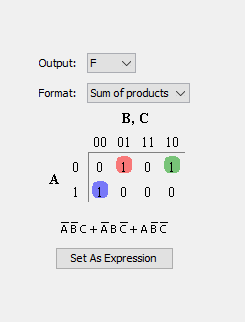
We can write F as a minterm expansion using the table entries which gives an output 1. Writing F as minterm expansion will yield,

***This is Standarad or Canonical Minterm expansion of F.***

We can verify this expression using Logisim by building a circuit for this function and verifying its truth table with the above table.

The truth table for the above circuit is,



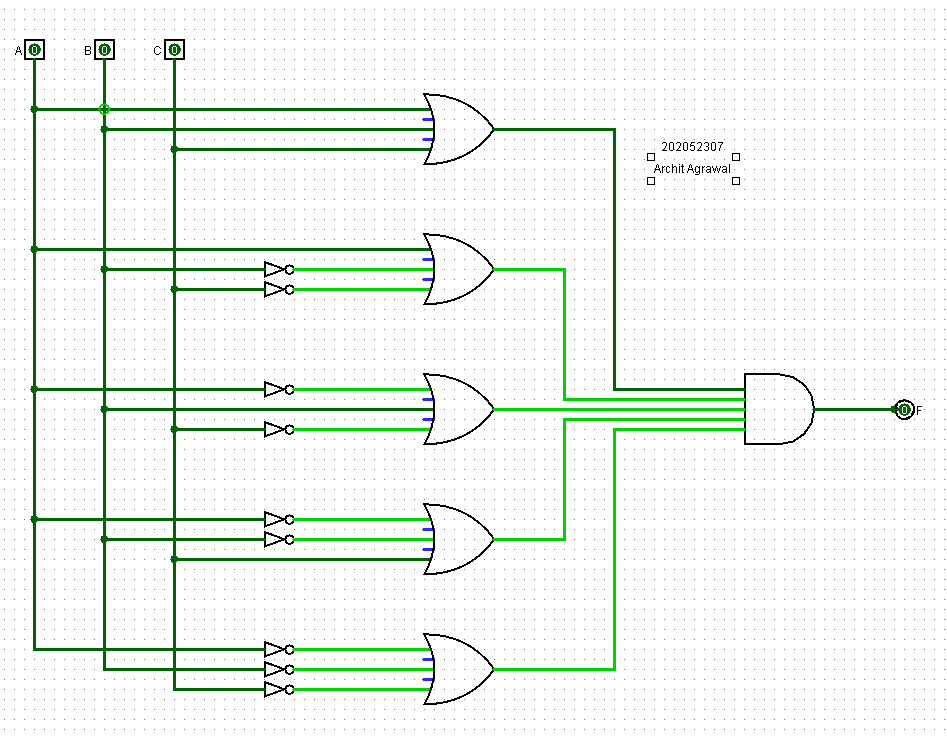


We can easily verify that the truth table is similar to the table we obtained in starting. Hence, the minterm expansion of F is verified.

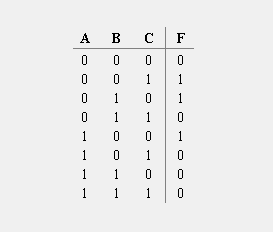
We can write F in another form known as maxterm expansion form using the entries that gives 0 output in the table. Writing F as maxterm expansion will yield,

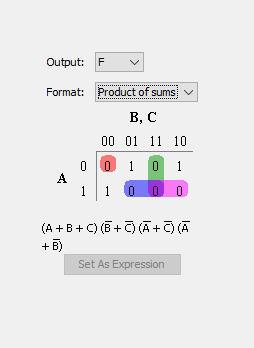
***This is Standarad or Canonical Maxterm expansion of F.***

We can verify this expression using Logisim by building a circuit for this function and verifying its truth table with the above table.



The truth table for the above circuit is,





We can easily verify that the truth table for above circuit is same as the table that we made initially. Hence, the maxterm expression of F is verified.

***Problems: Without the Simulations***

1. **A Boolean function f of two variables X and Y is defined as follows: f(0, 0) = f(1, 0) = f(1, 1) = 1; f(0, 1) = 0 Assuming complements of X and Y are not available, a minimum cost solution for realizing using only 2-input NOR gates and 2-input OR gates (each having unit cost) would have a total cost of?**

***Solution:***

Given, f(0, 0) = f(1, 0) = f(1, 1) = 1 and f(0, 1) = 0, the truth table for f(X, Y) is:

|  |  |  |
| --- | --- | --- |
| X | Y | f(X,Y) |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

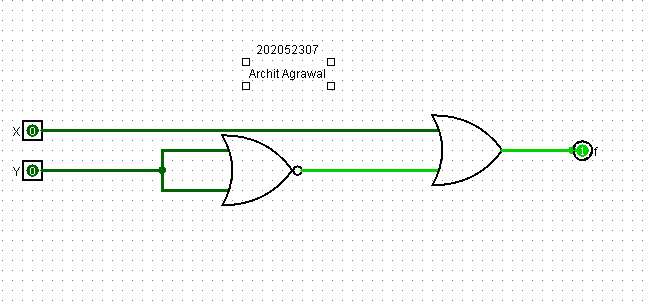
The K-map with possible pairings for the function f(X,Y) is:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Y** |  |  |
| **X** |  | **0** | **1** |
|  | **0** | 1 |  |
|  | **1** | 1 | 1 |

Following the rules of K-Map, the function f(X, Y) is:

f(X, Y) = X +

We will need one NOT gate(to get from Y) and one OR gate(to get X + from X, ). Since, we have only 2-input NOR and 2-input OR gates available, we will have to make NOT gate using NOR gate. The circuit for f(X, Y) is given below:



No. of NOR gates = 1

No. of OR gates = 1

Cost of NOR gate = 1

Cost of OR gate = 1