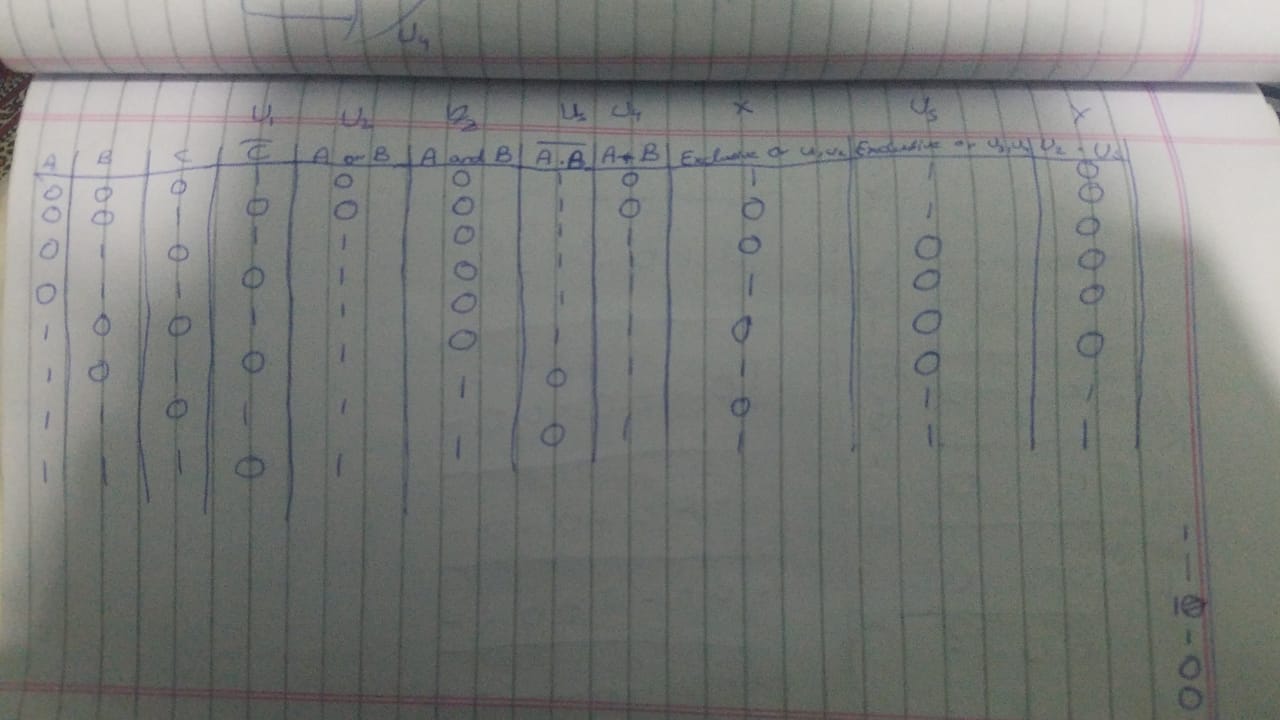
**Lab-2 Report**

**Question 1:**

The truth table of the given circuit is:



**Question 2:**

In the listing 4, following changes were made:

* First I removed “Logic” from line number 2,3,4,5 and 6.
* Then I removed the “,” from the end of line number 6 that was causing the syntax error.
* Then I used the “assign” statement in the start of line number 9 that was missing.
* Lastly, there was “&” operator missing inside the brackets before “(a^b)”, that was causing the error so I placed it there.

In the listing 5, following changes were made:

* First I used “reg” statement instead of “Logic” statement in line number 2, 3 and 4 and used “wire” statement in line number 5.
* Then there was a line missing that was “wire carry1;” This line was missing and we used it because we needed to get the output of carry and store them in a variable from our system Verilog code.
* In line number 7, there was “dut” statement missing after our system Verilog module name that is use to connect the system Verilog file to our test bench file.
* In line number 18, there was written a, b, c instead of a1, b1, c1 so I corrected it.
* In line number 20, there was written “c” instead of c1 so I corrected it.
* In line number 24, there was written “b” instead of b1 so I corrected it.
* In line number 30, there was written “a” instead of a1 so I corrected it.
* Before the “endmodule” statement, there was “end” statement missing so I placed it there.

**Question 3:**

The corrected code of listing 4 is:

module full\_adder(

input a,

input b,

input c,

output sum,

output carry);

assign sum = (a ^ b) ^ c;

assign carry = (a & b) | (c & (a ^ b));

endmodule

And the corrected code of listing 5 is:

module full\_adder\_tb();

reg a1;

reg b1;

reg c1;

wire sum1;

wire carry1;

full\_adder dut(

.a(a1),

.b(b1),

.c(c1),

.sum(sum1),

.carry(carry1)

);

initial

begin

a1 = 0; b1 = 0; c1 = 0;

#10

a1 = 0; b1 = 0; c1 = 1;

#10

a1 = 0; b1 = 1; c1 = 0;

#10

a1 = 0; b1 = 1; c1 = 1;

#10

a1 = 1; b1 = 0; c1 = 0;

#10

a1 = 1; b1 = 0; c1 = 1;

#10

a1 = 1; b1 = 1; c1 = 0;

#10

a1 = 1; b1 = 1; c1 = 1;

#10

$stop;

end

endmodule