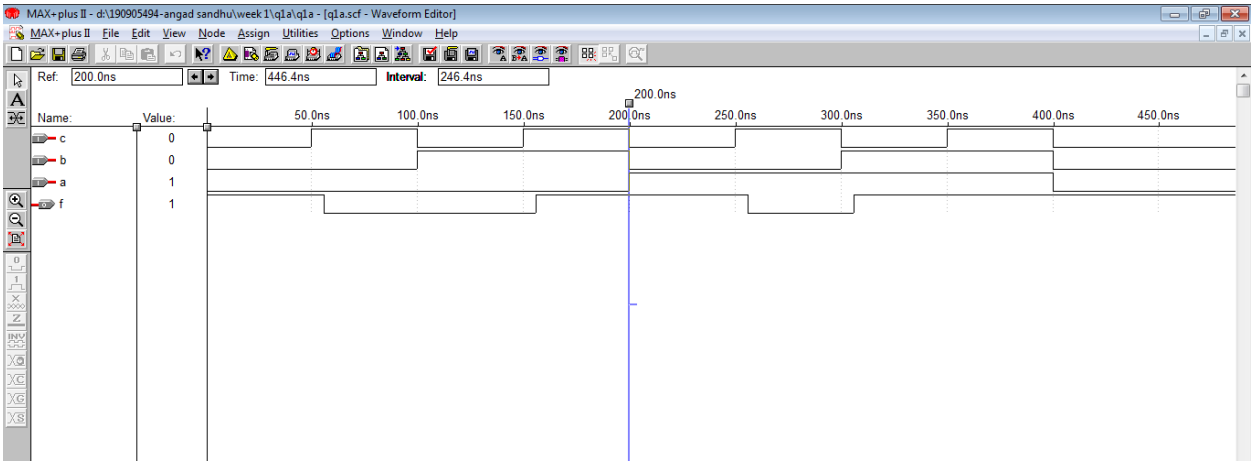
Q1)Write Verilog code to describe the following functions

f1 =ac'+bc+b'c'

f2 = (a+b'+c)(a+b+c')(a'+b+c')

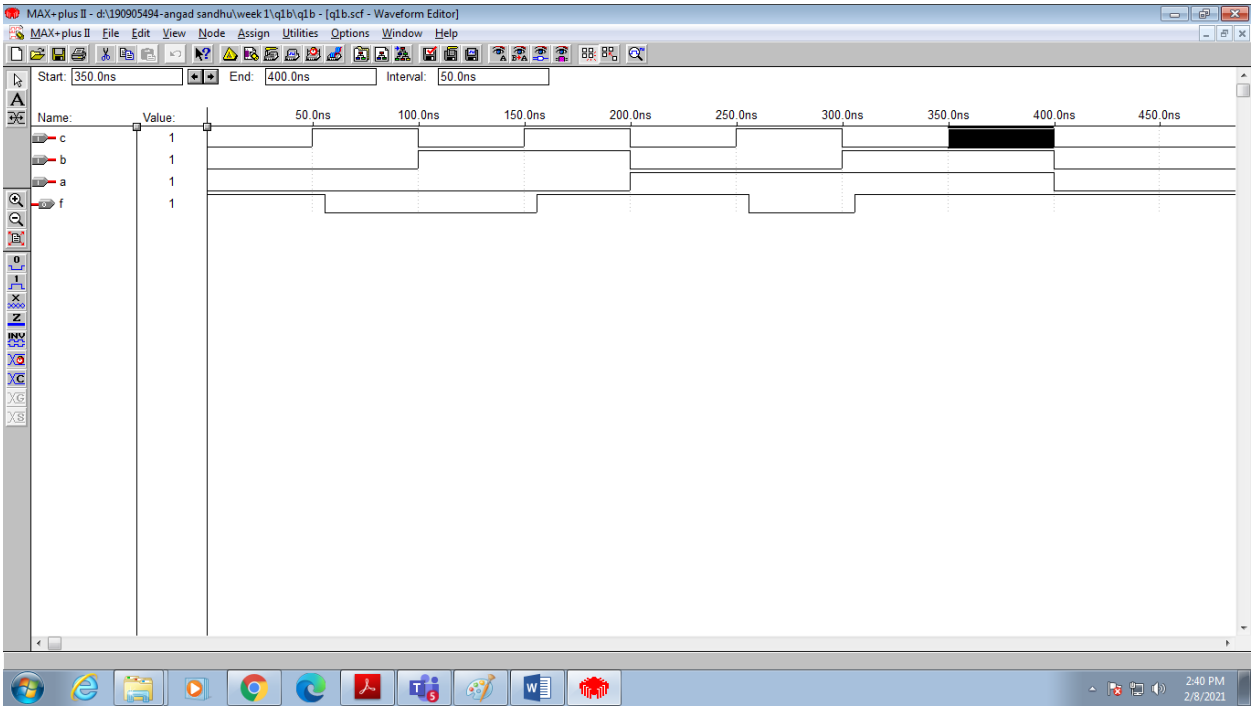
Ans 1) Part 1)

```
module q1a(a, b, c, f);
   input a, b, c;
   output f;

// creating compliments
// not(b_, b);
// not(c_, c);

// getting and gates
and(x1, a, ~c);
and(x2, b, c);
and(x3, ~b, ~c);
or(f, x1, x2, x3);
endmodule
```



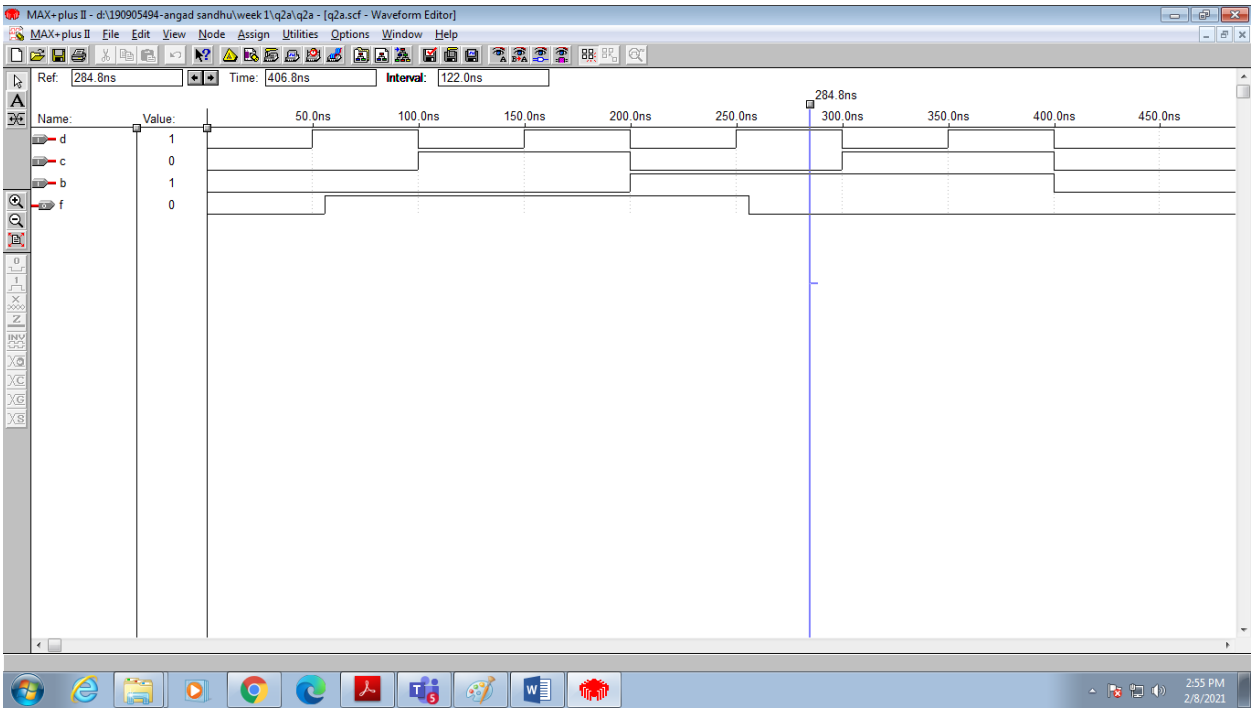
Ans 1) Part 2)

```
module q1b(a, b, c, f);
    input a, b, c;
    output f;

// getting and gates
or(x1, a, ~b, c);
or(x2, a, b, ~c);
or(x3, ~a, b, ~c);
and(f, x1, x2, x3);
endmodule
```



Q2) Part 1)

```
// equation to be minimized :
// f(A, B, C, D) = ?m(1, 3, 4, 9, 10, 12) + D(0, 2, 5, 11)

// the minimized term of the equation :
// Ans = B'D + B'C + BC'D'

// using continuous assignment statements
module q2a(a, b, c, d, f);
   input a, b, c, d;
   output f;

   assign f = (~b & d) | (~b & c) | (b & ~c & ~d);
endmodule
```



Q2) part 2)

// equation to be minimized:

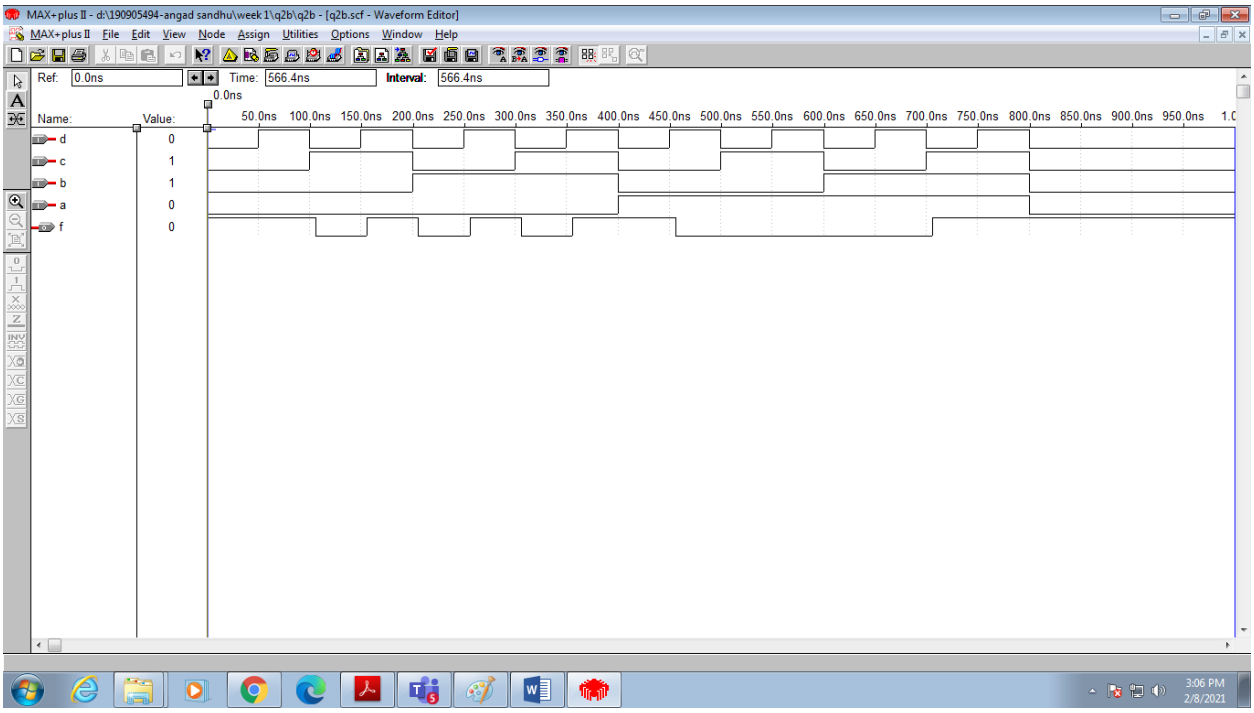
// f(A, B, C, D) = ?M(6, 9, 10, 11, 12) + D(2, 4, 7, 13)

// the minimized term of the equation:

// Ans = B'C'D' + ABC + A'D

// using continuous assignment statements

```
module q2b(a, b, c, d, f);
   input a, b, c, d;
   output f;

   assign f = (~a & d) | (~b & ~c & ~d) | (a & b & c);
endmodule
```



Q3)

```
// equation to be minimized :
// f(A, B, C, D) = ?M(2,6,8,9,10,11,14)

// the minimized term of the equation :
// Ans = A'C' + A'D + BC' + BD

// using continuous assignment statements
module q3(a, b, c, d, f);
   input a, b, c, d;
   output f;

        nand(x1, ~a, ~c);
        nand(x2, ~a, d);
        nand(x3, b, ~c);
        nand(x4, b, d);

        nand(f, x1, x2, x3, x4);
endmodule
```