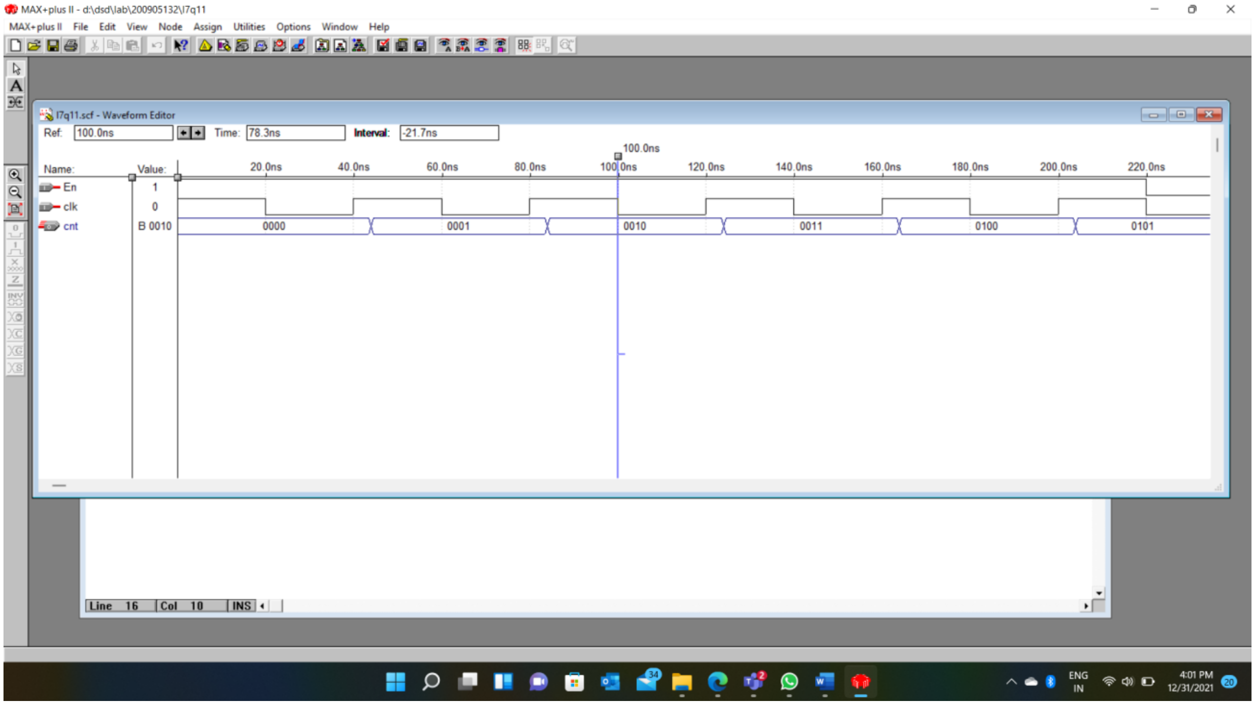
DSD LAB 7

1. Design and simulate the following counters

a) 4-bit synchronous up counter

```
module tFlipFlop(t,clock,q);
input t,clock;
output q;
reg q;
always@ (posedge clock)
if(t)
q<=~q;
endmodule

module l7q11(clk, En, cnt);
input clk, En;
output [3:0]cnt;
tFlipFlop stage0(En,clk,cnt[0]);
tFlipFlop stage1(En & cnt[0],clk,cnt[1]);
tFlipFlop stage2(En & cnt[1] & cnt[0],clk,cnt[2]);
tFlipFlop stage3(En & cnt[2] & cnt[1] & cnt[0],clk,cnt[3]);
endmodule
```

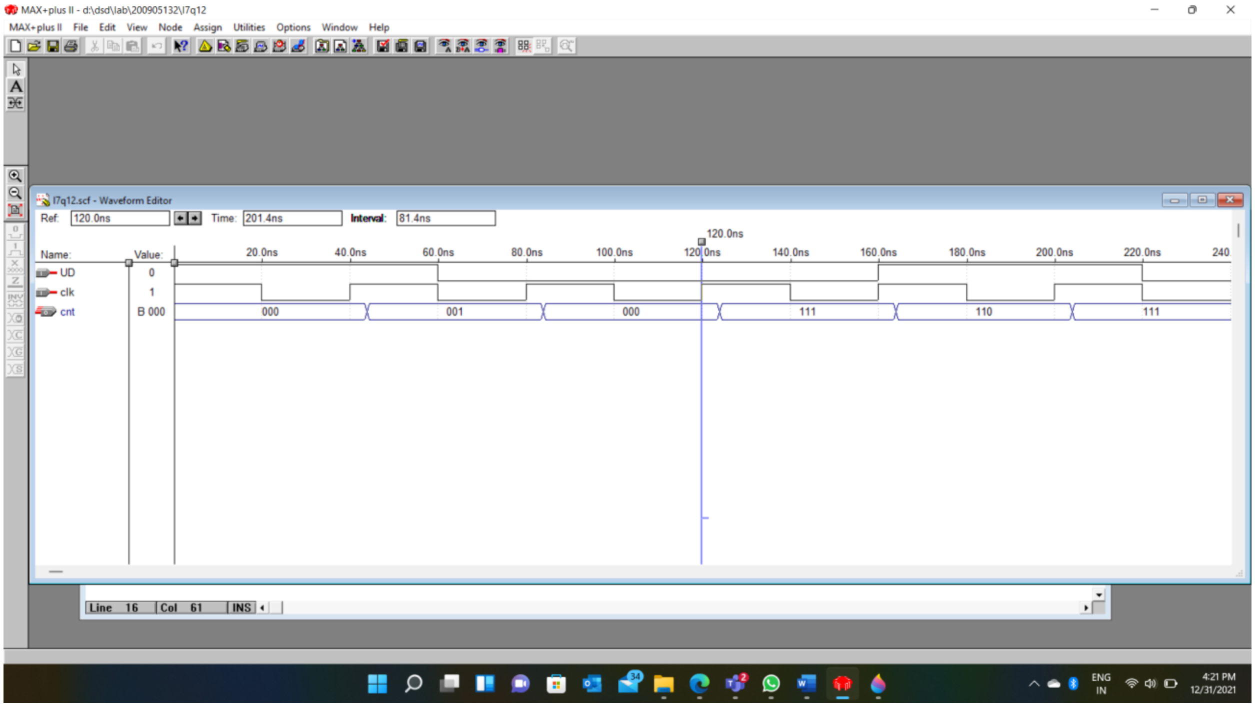

MAX+plus II - d:\dsd\lab\200905132\l7q11
MAX+plus II File Edit View Node Assign Utilities Options Window Help
l7q11.scf - Waveform Editor

Ref: 100.0ns Time: 78.3ns Interval: -21.7ns

Name:	Value:
En	1
clk	0
cnt	B 0010

b. 3 bit synchronous up/down counter with a control input up/down’. If up/down’ = 1, then the circuit should behave as an up counter. If up/down’ = 0, then the circuit should behave as a down counter.

```
module tFlipFlop(t,clock,q);

input t,clock;

output q;

reg q;

always@ (posedge clock)

if(t)

q<=~q;

endmodule


module l7q12(clk,UD, cnt);

input clk, UD;

output [2:0]cnt;

wire control;

tFlipFlop stage0(1,clk,cnt[0]);
tFlipFlop stage1((UD & cnt[0]) | (~UD & ~cnt[0]),clk,cnt[1]);

tFlipFlop stage2((UD & cnt[0] & cnt[1]) | (~UD & ~cnt[0] & ~cnt[0]),clk,cnt[2]);

endmodule
```



MAX+plus II - d:\dsd\lab\200905132\l7q12

l7q12.scf - Waveform Editor

Ref: 120.0ns Time: 201.4ns Interval: 81.4ns

Name:	Value:
UD	0
clk	1
cnt	B 000

Line 16 Col 61 INS