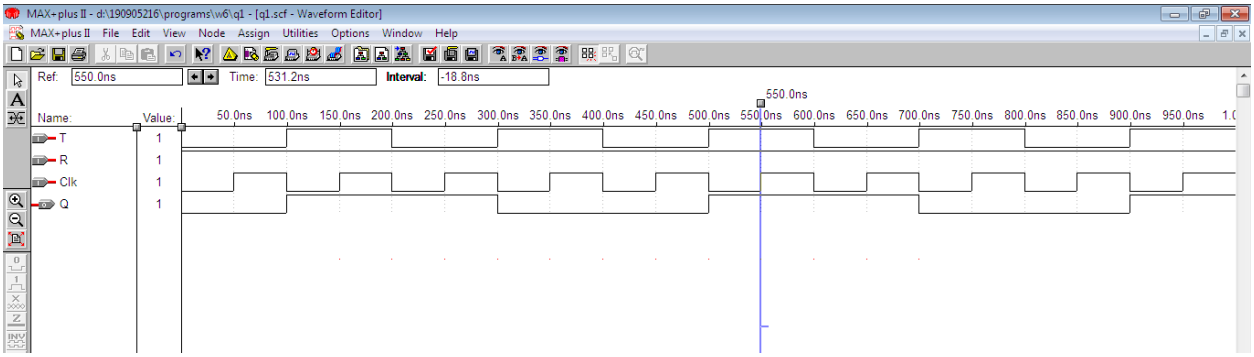
Week 6:

Q1:

```
module q1(T, Clk, R, Q);
input T,Clk, R;
output Q;
reg Q;
always @(negedge Clk or negedge R)
begin
        if(!R)
                Q<=0;
        else
                begin
                if(T)
                        Q<=~Q;
                else
                        Q<=Q;
                end
        end
endmodule
```

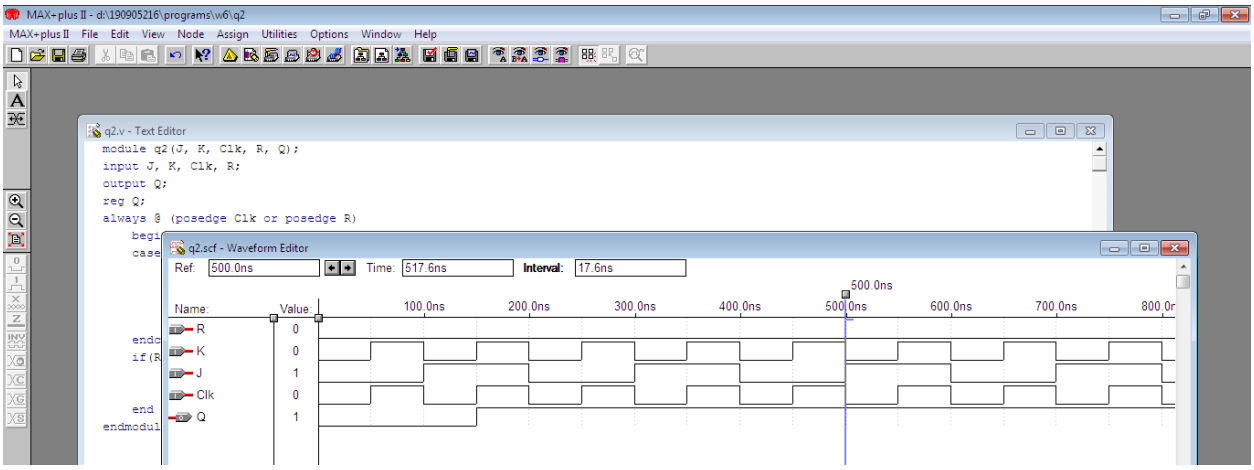
Output



Q2:

```
module q2(J, K, Clk, R, Q);
input J, K, Clk, R;
output Q;
reg Q;
always @ (posedge Clk or posedge R)
        begin
   case({J, K})
                0: Q <= Q;
                1: Q <= 0;
                2: Q <= 1;
                3: Q <= ~Q;
        endcase
        if(R == 1)
                                Q <= 0;

        end
endmodule
```



Q3:

Part 1:

```
module DFlipFlop(D, clock,reset, q);
 input D, clock, reset;
output q;
 reg q;
 always @ (posedge clock)
 begin
 if(reset)
 q <= 0;
 else if(D)
 q <= 1;
else
q <= 0;
 end
endmodule

module twobitc(clock, reset, q);
input clock, reset;
output [0:1]q;
wire [0:1]q;
DFlipFlop d1(~q[1], clock, reset, q[0]);
DFlipFlop d2(q[0], clock, reset, q[1]);
endmodule

module dec2to4(W,En,Y);
input[1:0]W;
input En;
output [0:3]Y;
```

```
reg [0:3]Y;
always@(W or En)
begin
if(En==1)
case(W)
 0: Y=4'b1000;
 1: Y=4'b0100;
 2: Y=4'b0010;
3: Y=4'b0001;
endcase
else
 Y=4'b0000;
end
endmodule


module q3p1(Clock, Reset, Q);
input Clock, Reset;
output [3:0] Q;
wire [3:0] Q;
wire [1:0] temp;
twobitc c1(Clock, Reset, temp);
dec2to4 d1(temp, 1'b1, Q);
endmodule
```

Part 2:

```
module DFlipFlop(D, clock, reset, q);
 input D, clock,reset;
output q;
reg q;

always @ (posedge clock)
                begin
                        if(reset)
                                q <= 0;
                        else if(D)
                                q <= 1;
                        else
                                q <= 0;
                end
endmodule

module q3p2(clock, reset, q);
input clock, reset;
output [0:4]q;
        wire [0:4]q;

        DFlipFlop d1(~q[4], clock, reset, q[0]);
        DFlipFlop d2(q[0], clock, reset, q[1]);
        DFlipFlop d3(q[1], clock, reset, q[2]);
        DFlipFlop d4(q[2], clock, reset, q[3]);
        DFlipFlop d5(q[3], clock, reset, q[4]);
```

```
endmodule
```