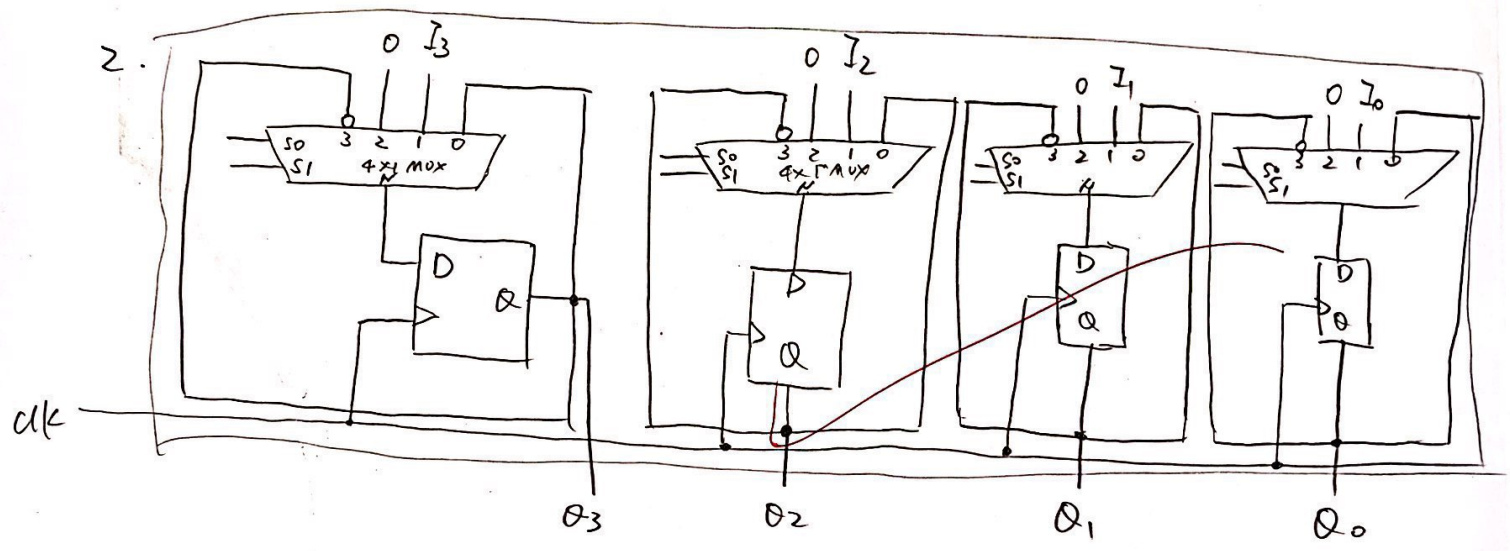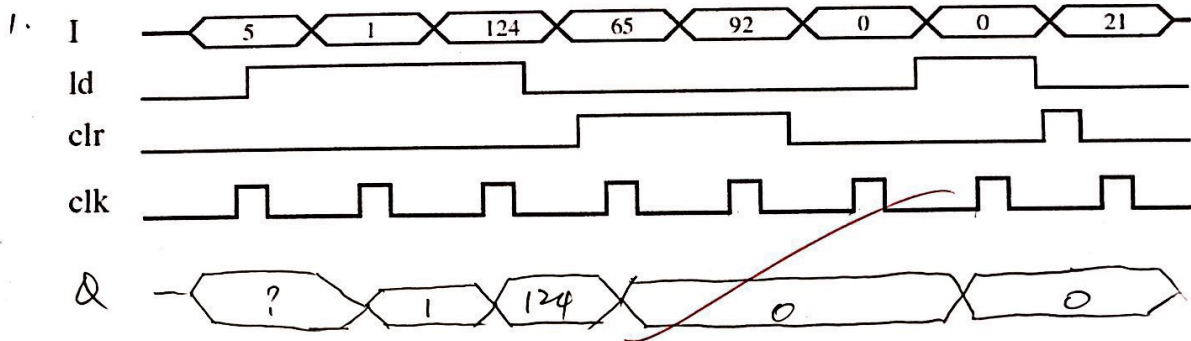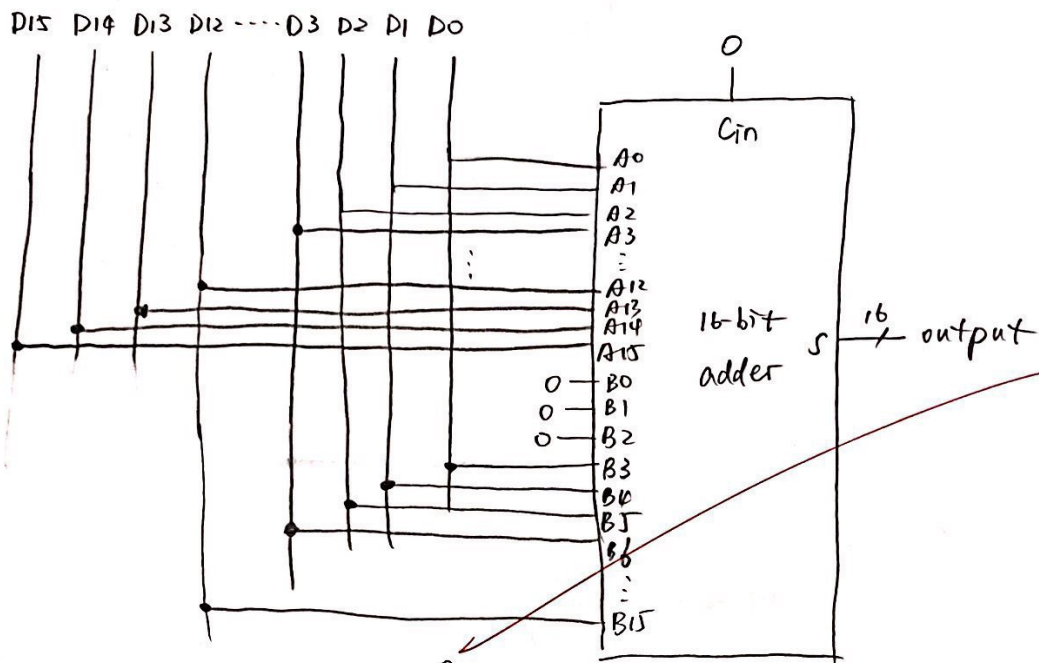VE270 HW6
He Yifei 5173709101.96

1.

2.

3. We need to shift the number 3 digits to the left and add itself.

DI5 DI4 DI3 DI2 ···· D3 D2 DI D0

C_{in} — 0

16-bit adder

A0
A1
A2
A3
⋮
A12
A13
A14
A15

0 — B0
0 — B1
0 — B2
B3
B4
B5
B6
⋮
B15

S — 16 / output

4. (a)

$0 \uparrow 8$
D

clear

8-bit counter(up)

Clk

$Q_7 Q_6 Q_5 Q_4 Q_3 Q_2 Q_1 Q_0$

output

(b)

0 1 1 0 0 0 1 0

D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0

load

8 bit counter (down)
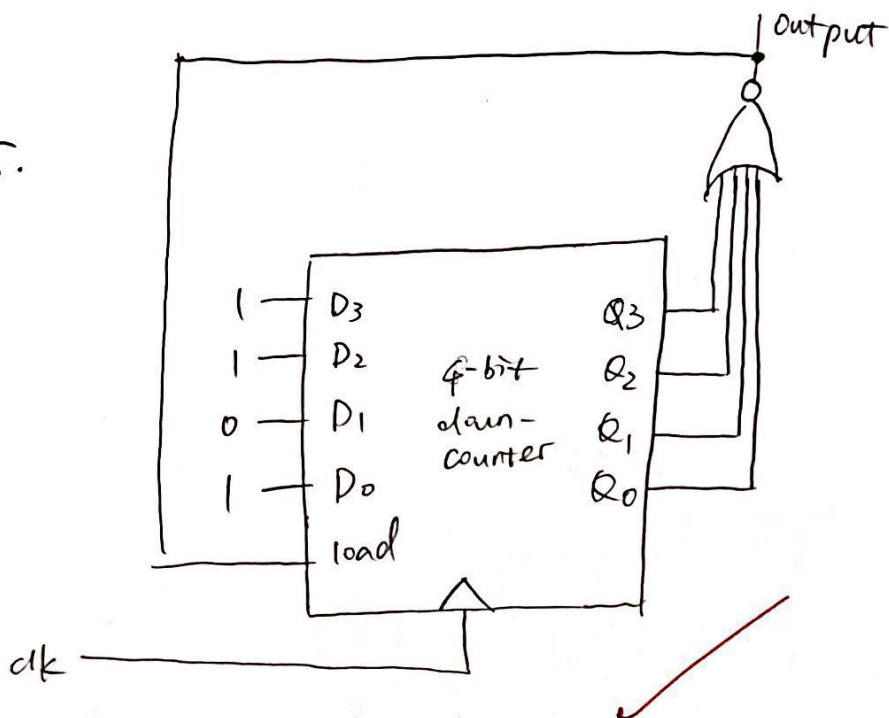
clk

$Q_7 Q_6 Q_5 Q_4 Q_3 Q_2 Q_1 Q_0$
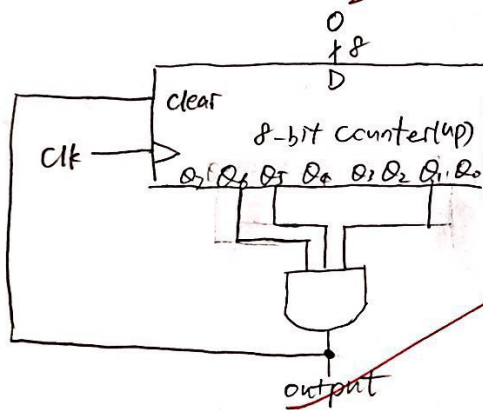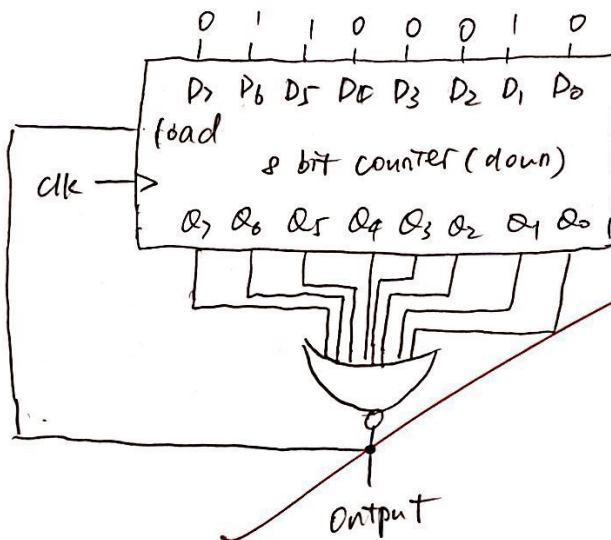
Output

5.

6.

```
module usr(Q, SI, D, Sh, L, clk);
input [3:0] D;
output [3:0] Q;
input Sh, L, clk, SI;
reg [3:0] Q;

always @ (posedge clk)
begin
    if(Sh==0 && L==1) Q<=D;
    else if(Sh==1) begin Q[2:0]<=Q[3:1];Q[3]<=SI;end
    else if(Sh==0 && L==0) Q<=Q;
end
endmodule
```

```
module sim();
parameter half_period=20;
reg [3:0] D;
reg Sh, L, clk, SI;
wire [3:0] Q;

usr usrr(Q, SI, D, Sh, L, clk);
initial begin
    #0   clk = 0; D = 4'b1111; Sh=0; L=0; SI=0;
    #80 L=1;
    #80 Sh=1;
    #80 SI=1;
    #120 SI=0;
    #20 Sh=1;SI=0;
    #100 L=0; Sh=0;
end
always #half_period clk=~clk;
initial #1000 $stop;
endmodule
```