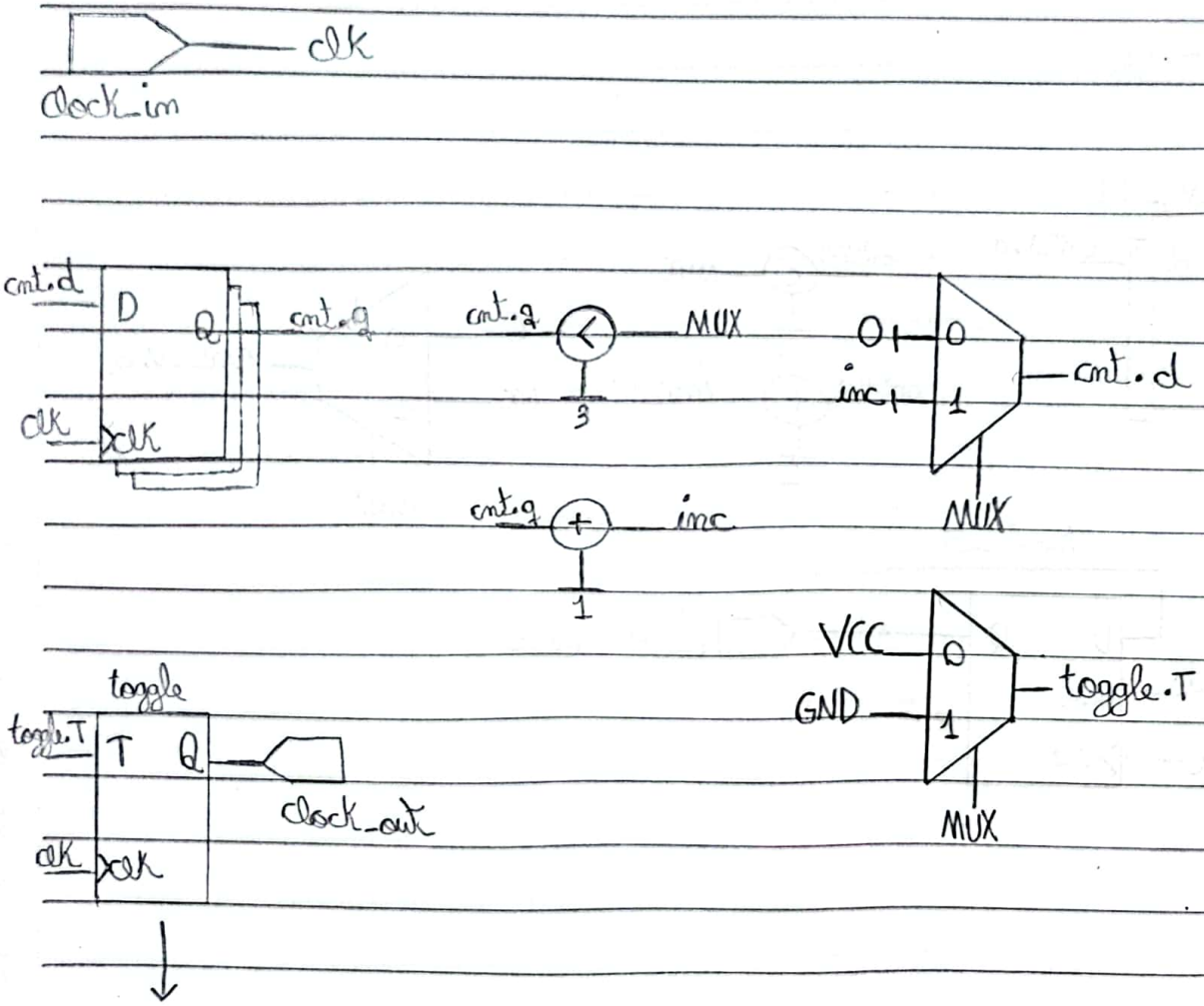
Prox-estado	01 S_1	11 S_2	00 S_0	01 S_1	11 S_2	00 S_0	01 S_1	11 S_2	00 S_0	01 S_1
estado-atual	00 S_0	01 S_1	10 S_2	00 S_0	01 S_1	11 S_2	00 S_0	01 S_1	11 S_2	00 S_0
S_0	1	0	0	1	0	0	1	0	0	1
S_1	0	1	0	0	1	0	0	1	0	0
S_2	0	0	1	0	0	1	0	0	1	0
saída	0	1	0	0	1	0	0	1	0	0
clock										

Diagrama de Temporização referente a (2º) e (3º) questão

Estm	00 S_0	01 S_1	11 S_2	00 S_0	01 S_1	11 S_2	00 S_0	01 S_1	11 S_2	00 S_0
S_0	0	0	0	1	0	0	1	0	0	1
S_1	0	1	0	0	1	0	0	1	0	0
S_2	0	0	1	0	0	1	0	0	1	0
saída	1	0	0	1	0	0	1	0	0	1
clock										

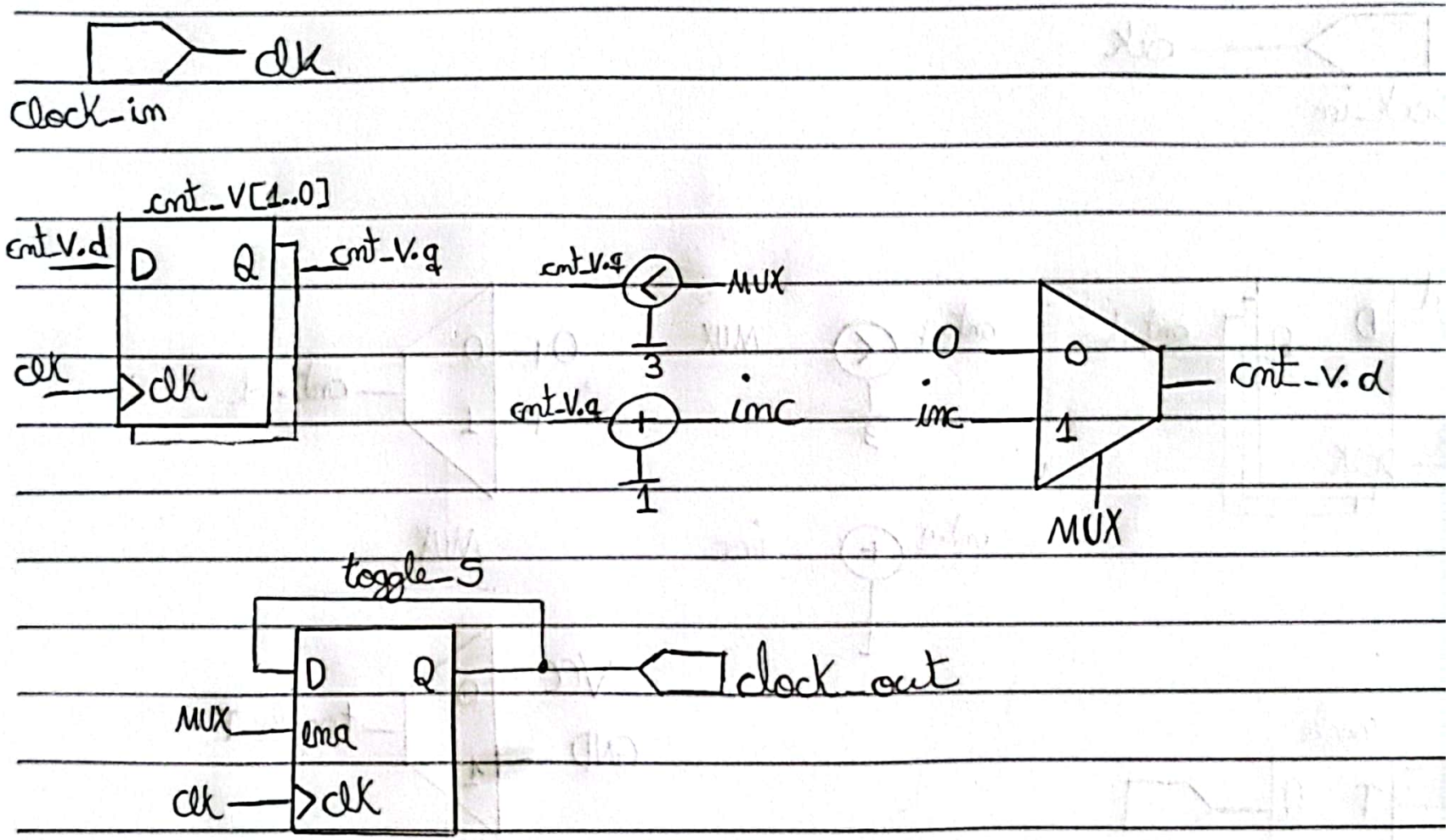
Diagrama de Temporização referente a (1º) questão.

4º) Esquemático RTL:



- OBS: Pode ser substituido por um enable no toggle.

5º e 6º O esquemático dessas duas questões são iguais:



```
1   SUBDESIGN Questao7(
2       A[3..1], B[3..1]   :  INPUT;
3       saida              :  OUTPUT;
4   )
5
6     VARIABLE
7       aux[3..1]    :  NODE;
8     BEGIN
9
10      FOR i IN 1 to 3 GENERATE
11
12        IF i == 1 GENERATE
13          aux[i] = A[i] OR B[i];
14        END GENERATE;
15
16        IF i == 2 GENERATE
17          aux[i] = A[i] NAND B[i];
18        END GENERATE;
19
20        IF i == 3 GENERATE
21          aux[i] = A[i] NOR B[i];
22        END GENERATE;
23
24      END GENERATE;
25
26      saida  = aux[1] AND aux[2] AND aux[3];
27
28    END;
```

```vhdl
LIBRARY IEEE;
USE IEEE.std_logic_1164.ALL;
USE IEEE.numeric_std.ALL;

ENTITY Questao8 IS

    PORT(
        A       :   IN      std_logic_vector(3 DOWNTO 1);
        B       :   IN      std_logic_vector(3 DOWNTO 1);
        saida   :   OUT std_logic
    );

END ENTITY;

ARCHITECTURE arc_quest8 OF Questao8 IS

    SIGNAL aux  :   std_logic_vector(3 DOWNTO 1);
BEGIN

    F1 : FOR i IN aux'LOW TO aux'HIGH GENERATE
    BEGIN

        operacao1 : IF(i = 1) GENERATE
        BEGIN

            aux(i) <= A(i) OR B(i);

        END GENERATE operacao1;

        operacao2 : IF(i = 2) GENERATE
        BEGIN

            aux(i) <= A(i) NAND B(i);

        END GENERATE operacao2;

        operacao3 : IF(i = 3) GENERATE
        BEGIN

            aux(i) <= A(i) NOR B(i);

        END GENERATE operacao3;

    END GENERATE F1;

    saida <= aux(1) AND aux(2) AND aux(3);

END ARCHITECTURE;
```

```
 1  module questao9
 2  (
 3      input [3:1]A,
 4      input [3:1]B,
 5      output Saida
 6  );
 7
 8      wire [3:1]B1;
 9      wire WireAnd0;
10      genvar i;
11
12      generate
13         for(i = 1 ; i < 4 ; i = i + 1) begin : L1
14             if(i == 1) begin
15
16               assign B1[i] = (A[i] || B[i]);
17             end
18
19             else if(i == 2) begin
20
21               assign B1[i] = ~(A[i] && B[i]);
22             end
23
24             else if(i == 3) begin
25
26               assign B1[i] = ~(A[i] || B[i]);
27             end
28         end
29      endgenerate
30
31      assign WireAnd0  = B1[1] & B1[2] & B1[3];
32
33      assign Saida = WireAnd0;
34
35  endmodule
36
```