

# TP3 EN 1B

# CHRONOMETRE

# 2021-2022

**Oliver BELLIARD & Fahd RAFIK**

## Préparation

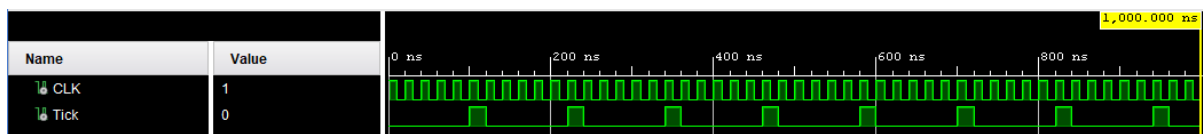
**6.A)** Le rapport de division sera de  $100 \cdot (10^6) / 1000 = 10^5$

**6.B)** On doit compter jusqu'à  $10^7$  donc  $\sim 2^{17}$ , nous avons à priori besoin de 17 bascules D de sortie de 1 bit ou 1 de 20 bits.

**6.D)** On peut fixer une valeur très petite comme condition pour commencer pour générer un tick et nous assurer que cette fonctionnalité fonctionne. Ensuite on peut simuler le signal de 100MHz et s'assurer que le code marche même pour de grandes valeurs de Q (la variable de comptage).

## Simulation 6

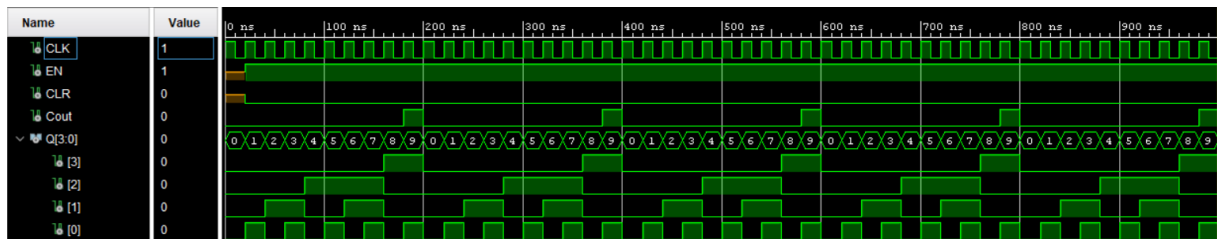
Pour tester notre TICK\_1ms nous avons essayé pour un tick tous les 5 fronts montants d'horloge. Nous obtenons :



Nous avons bien un "tick" tous les 5 fronts d'horloge, nous pouvons maintenant procéder à remplacer la valeur max de la variable de comptage par la valeur définitive.

## Simulation 8

Nous avons fixé N à 10 pour la simulation. On remarque le bon fonctionnement de Cnt0toN\_EN :



## Annexes

Vous trouverez en annexe la prépa Prep6.C (le fichier TICK\_1ms.vhd), Disp4D complété ainsi que son schéma et Cnt0toN ainsi que son fichier de simulation pour mieux comprendre la capture de la simulation.

# Annexe 1 :

## Préparation Prep6.C : TICK\_1ms.vhd

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity TICK_1ms is
    Port ( CLK : in STD_LOGIC;
           Tick : out STD_LOGIC);
end TICK_1ms;

architecture Behavioral of TICK_1ms is
    constant Ndiv: integer := 99999;
    signal Q: integer range 0 to Ndiv:= 0;
begin

    process(CLK)begin
        if rising_edge(CLK)then
            if(Q = Ndiv)then
                Q <= 0;
            else
                Q <= Q+1;
            end if;
        end if;
    end process;

    Tick<= '1' when Q = Ndiv else '0';
end Behavioral;
```

# Annexe 2 :

## Disp4D.vhd complété

```
-----
-- Company:
-- Engineer:
--
-- Create Date: 23.11.2021 11:28:11
-- Design Name:
-- Module Name: Disp4D - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
-----
```

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
```

```
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
use IEEE.NUMERIC_STD.ALL;
```

```
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;
```

```
entity Disp4D is
    Port ( A : in STD_LOGIC_VECTOR (3 downto 0);
          B : in STD_LOGIC_VECTOR (3 downto 0);
          C : in STD_LOGIC_VECTOR (3 downto 0);
          D : in STD_LOGIC_VECTOR (3 downto 0);
          CLK : in STD_LOGIC;
          S : out STD_LOGIC_VECTOR (6 downto 0);
          ENAN : out STD_LOGIC_VECTOR (3 downto 0));
end Disp4D;
```

```
architecture Behavioral of Disp4D is
    signal Sel: STD_LOGIC_VECTOR(1 downto 0);
    signal EN_1ms: STD_LOGIC;
begin

    u1 : entity work.Cmpt0to3
        port map (CLK=>CLK, EN=>EN_1ms, N=>Sel);

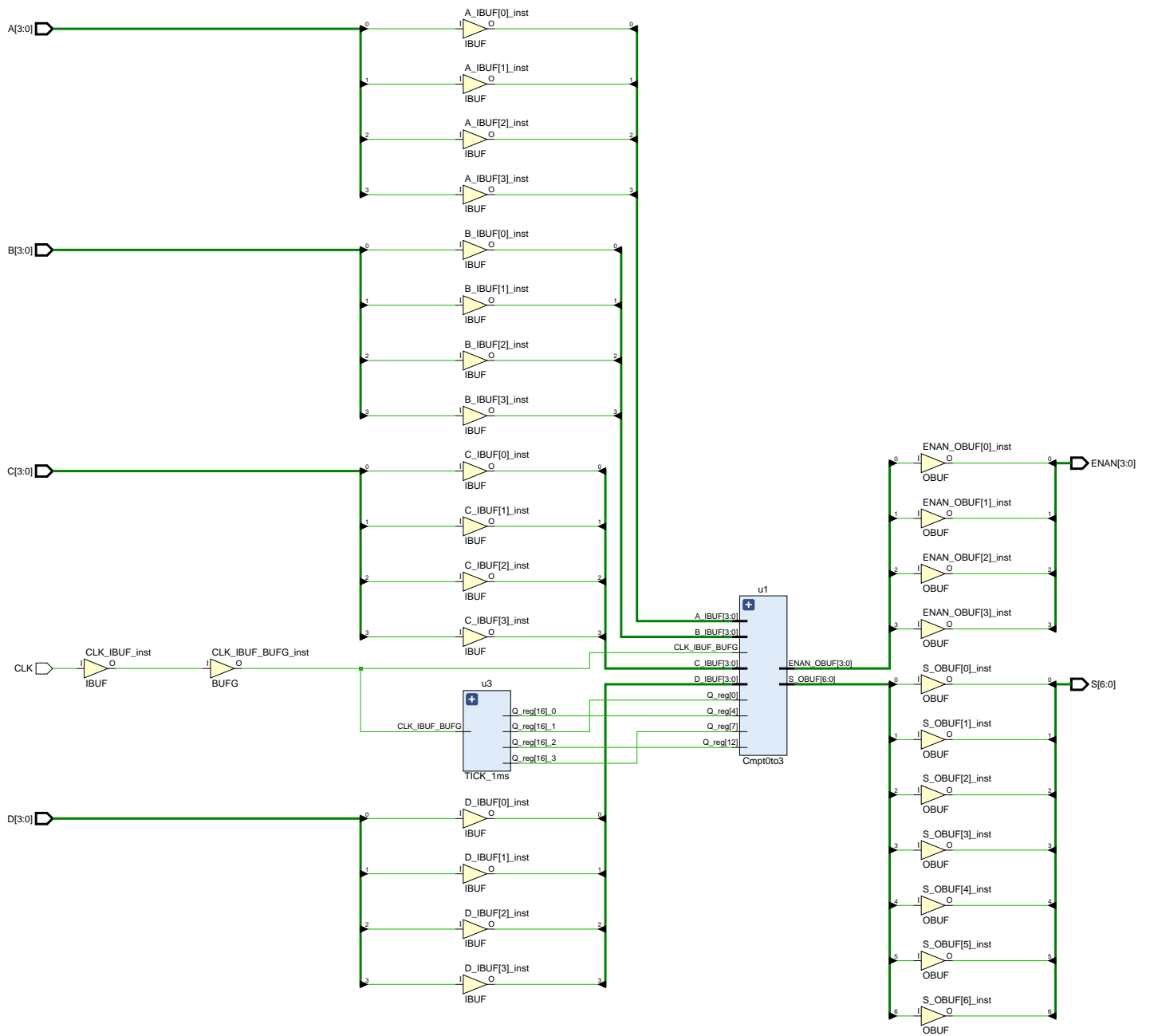
    u2 : entity work.Disp1of4Digits
        port map (A=>A, B=>B, C=>C, D=>D, N=>Sel, S=>S, ENAN=>ENAN);

    u3 : entity work.TICK_1ms
        port map (CLK=>CLK, Tick=>EN_1ms);

end Behavioral;
```

# Annexe 3 :

## Schéma de la description Disp4D





# Annexe 4 :

## Cnt0toN\_EN complété

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity Cnt0toN_EN is
    generic ( N : natural := 10 );

    Port ( CLK : in STD_LOGIC;
          EN : in STD_LOGIC;
          CLR : in STD_LOGIC;
          Cout : out STD_LOGIC;
          Q : out STD_LOGIC_VECTOR (3 downto 0));
end Cnt0toN_EN;

architecture Behavioral of Cnt0toN_EN is
    signal M: natural range 0 to N-1 := 0;

begin
    process(CLK)begin
        if rising_edge(CLK)then
            --évolution de M
            if (CLR='1' or M=N-1) then
                M <= 0;
            else
                if (EN='1') then
                    M <= M+1;
                else
                    M <= M;
                end if;
            end if;
        end if;
    end process;
    --recopie de M sur la sortie Q
    Q <= STD_LOGIC_VECTOR(to_unsigned(M,4));
    Cout <= '1' when M=N-1 else '0';

end Behavioral;
```

# Annexe 5 :

## Cnt0toN\_EN\_tb (fichier de simulation)

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Cnt0toN_EN_tb is

end Cnt0toN_EN_tb;

architecture Behavioral of Cnt0toN_EN_tb is
    component Cnt0toN_EN
        generic ( N : natural := 10 );
    Port ( CLK : in STD_LOGIC;
          EN : in STD_LOGIC;
          CLR : in STD_LOGIC;
          Cout : out STD_LOGIC;
          Q : out STD_LOGIC_VECTOR (3 downto 0));
    end component;

    signal CLK, EN, CLR, Cout : STD_LOGIC;
    signal Q : STD_LOGIC_VECTOR (3 downto 0);

begin
    UTT: Cnt0toN_EN generic map (N=>10)
        port map (CLK=>CLK, EN=>EN, CLR=>CLR, Cout=>Cout, Q=>Q);

    process
    begin
        CLK <= '1';
        wait for 10 ns;
        CLK <= '0';
        wait for 10 ns;
        EN <= '1';
        CLR <= '0';
    end process;
end Behavioral;

```