Luis Umana
801018765

ECGR 4146: Introduction to VHDL
Lab 6: 8-bit Counter

VHDL Code for Counter (0 to m-1)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;
entity MCounter is
    Port ( clk : in STD_LOGIC;
           reset : in STD_LOGIC;
           m : in STD_LOGIC_VECTOR(3 downto 0);
           q : out STD_LOGIC_VECTOR(3 downto 0)
           );
end MCounter;
architecture Behavioral of MCounter is
 signal R_Register: unsigned(3 downto 0);
 signal R_Next: unsigned(3 downto 0);
begin
   process(clk,reset)
   begin
      if (reset='1') then
         R_Register <= (others=>'0');
      elsif (clk'event and clk='1') then
         R_Register <= R_Next;
      end if;
   end process;
   R_next <= (others=>'0') when R_Register=(unsigned(m)-1) else
             R_Register + 1;
   q <= std_logic_vector(R_Register);
end Behavioral;
```

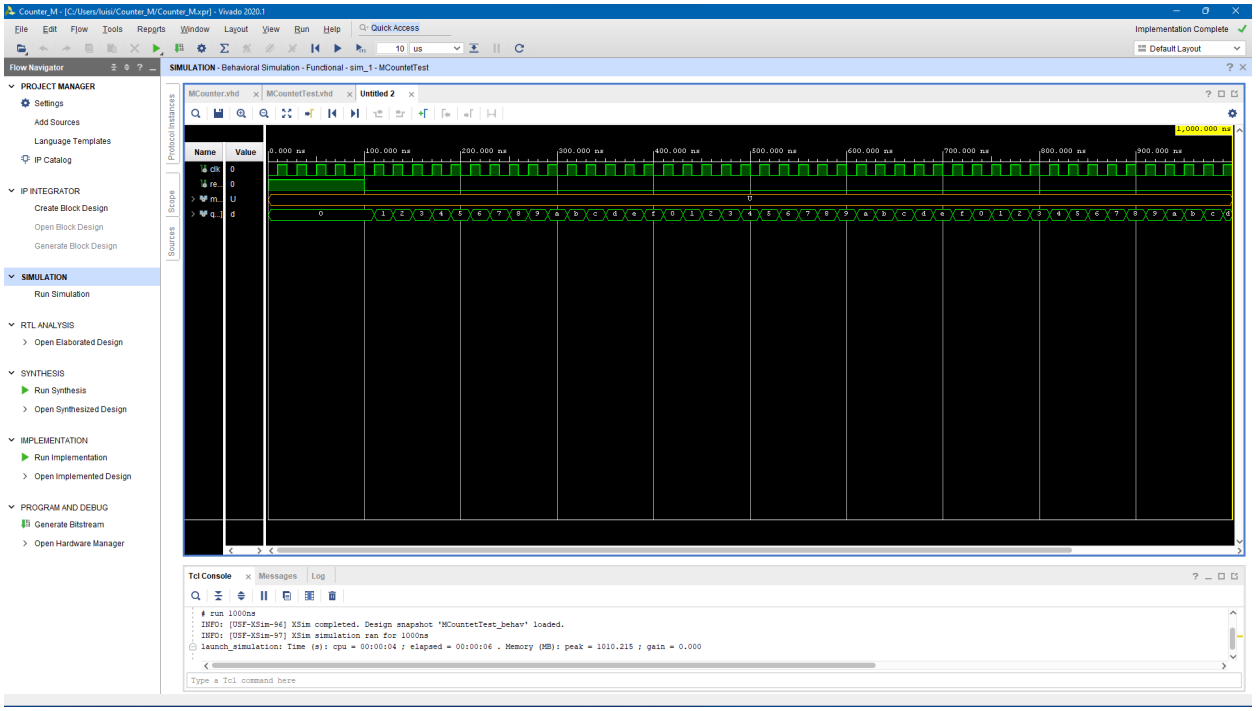
VHDL Testbench Code for Counter (0 to m-1)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;
entity MCountetTest is
end MCountetTest;
architecture Behavioral of MCountetTest is
component MCounter is
    Port ( clk : in STD_LOGIC;
           reset : in STD_LOGIC;
           m : in STD_LOGIC_VECTOR(3 downto 0);
           q : out STD_LOGIC_VECTOR(3 downto 0)
           );
end component;
signal clk :std_logic;
signal reset :std_logic;
signal m:std_logic_vector(3 downto 0);
signal q:std_logic_vector(3 downto 0);
begin
uut:MCounter port map
```

Luis Umana
801018765

```
(clk=>clk,reset=>reset,m=>m,q=>q);
clock_process :process
begin
     clk <= '0';
     wait for 10 ns;
     clk <= '1';
     wait for 10 ns;
end process;
stim_proc: process
begin
    reset <= '1';
    wait for 100 ns;
    reset <= '0';
  wait;
end process;
end Behavioral;
```

Simulation Waveform of Half Adder



VHDL Code for Counter (n to m-1)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity Counter_M_N is
 generic (
            M : integer := 15;
```

Luis Umana
801018765

```
            N : integer := 8
    );
    Port ( clk : in STD_LOGIC;
           reset : in STD_LOGIC;
           Tick : out STD_LOGIC;
           Counting : out STD_LOGIC_VECTOR(N-1 downto 0)
           );
end Counter_M_N;
architecture Behavioral of Counter_M_N is
signal R_Register: std_logic_vector(N-1 downto 0);
signal R_Next: std_logic_vector(N-1 downto 0);
begin
   process(clk,reset)
   begin
      if (reset='1') then
         R_Register <= "00000110";
      elsif (clk'event and clk='1') then
         R_Register <= R_Next;
      end if;
   end process;
   R_next <= "00000110" when R_Register=(M-1) else
             R_Register + 1;
     Counting <= std_logic_vector(R_Register);
    Tick <= '1' when R_Register = (M-1) else '0';
end Behavioral;
```

VHDL Testbench Code for Counter (n to m-1)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;
entity Counter_M_NTest is
end Counter_M_NTest;
architecture Behavioral of Counter_M_NTest is
constant N: integer := 8;
signal clk :std_logic;
signal reset :std_logic;
signal Tick:std_logic;
signal Counting:std_logic_vector(N-1 downto 0);
begin
p0:entity work.Counter_M_N(Behavioral) port map
(clk=>clk,reset=>reset,Tick=>Tick,Counting=>Counting);
clock_process :process
begin
     clk <= '0';
     wait for 50 ns;
     clk <= '1';
     wait for 50 ns;
end process;
stim_proc: process
begin
    reset <= '1';
    wait for 50 ns;
    reset <= '0';
  wait;
end process;
```

Luis Umana
801018765

```
end Behavioral;
```

Simulation Waveform of Counter (n to m-1)