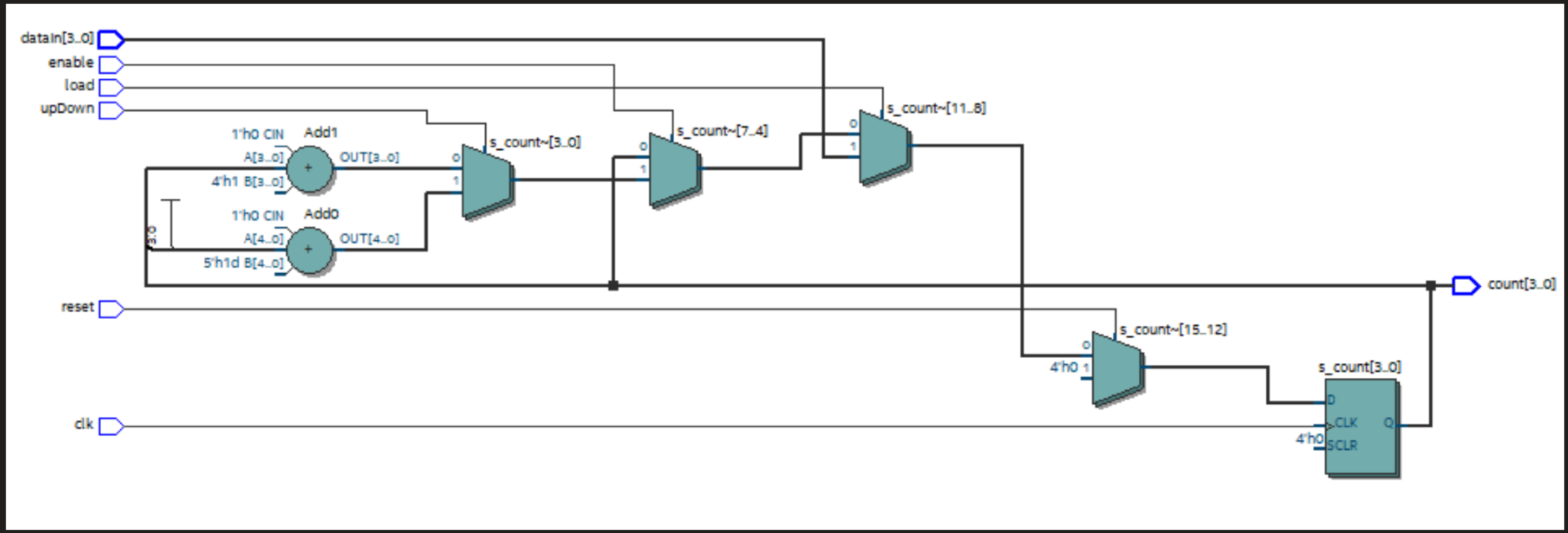
Synchronous 4 bit upDown binary counter

Tuesday, April 13, 2021 4:08 PM

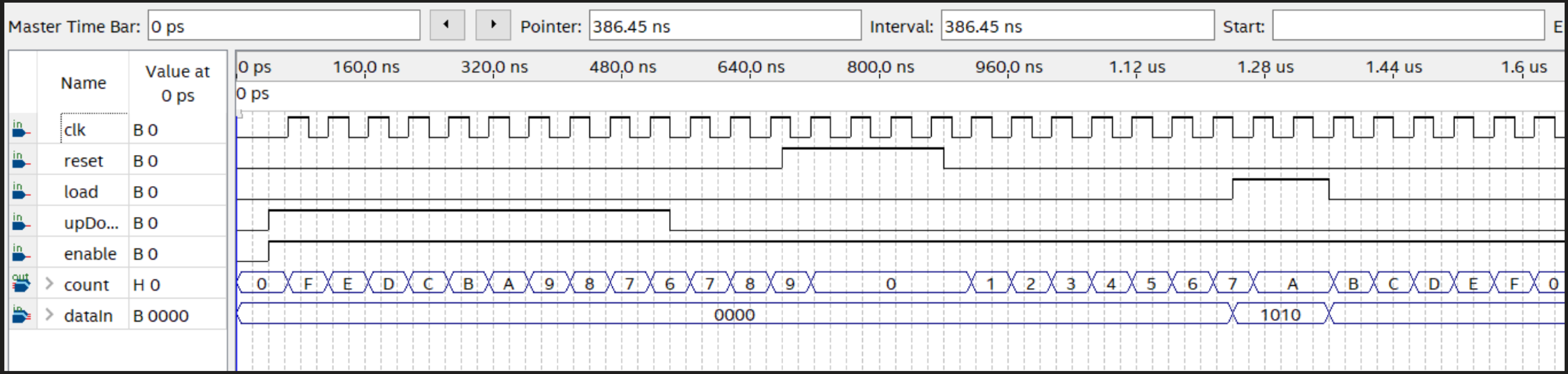
```vhdl
library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.NUMERIC_STD.all;
entity CounterUpDown4 is
      port(   clk : in  std_logic;
               reset : in  std_logic;
              upDown : in  std_logic;
                load : in std_logic;
              enable : in std_logic;
              dataIn : in std_logic_vector(3 downto 0);
               count : out std_logic_vector(3 downto 0));
      end CounterUpDown4;
architecture Behavioral of CounterUpDown4 is
signal s_count : unsigned(3 downto 0);
begin
      process( clk)
            begin
            if( rising_edge(clk) ) then
                  if (reset = '1') then s_count <= (others => '0');
                  elsif (load = '1') then s_count <= unsigned(dataIn);
                  elsif (enable = '1') then
                        if (upDown = '1') then
                              s_count <= s_count - 1;
                        else
                              s_count <= s_count + 1;
                        end if;
                  end if;
            end if;

      end process;
      count <= std_logic_vector(s_count);
end Behavioral;
```



Notice:

- Reset initialization
- The Load/Enable/Count flow in the process



Master Time Bar: 0 ps

Pointer: 386.45 ns

Interval: 386.45 ns

Start:

Name	Value at 0 ps
clk	B 0
reset	B 0
load	B 0
upDo...	B 0
enable	B 0
count	H 0
dataIn	B 0000