

Jiacheng Zhang
Lab 5 Report
ECE 2031 L07
18 Feb 2022

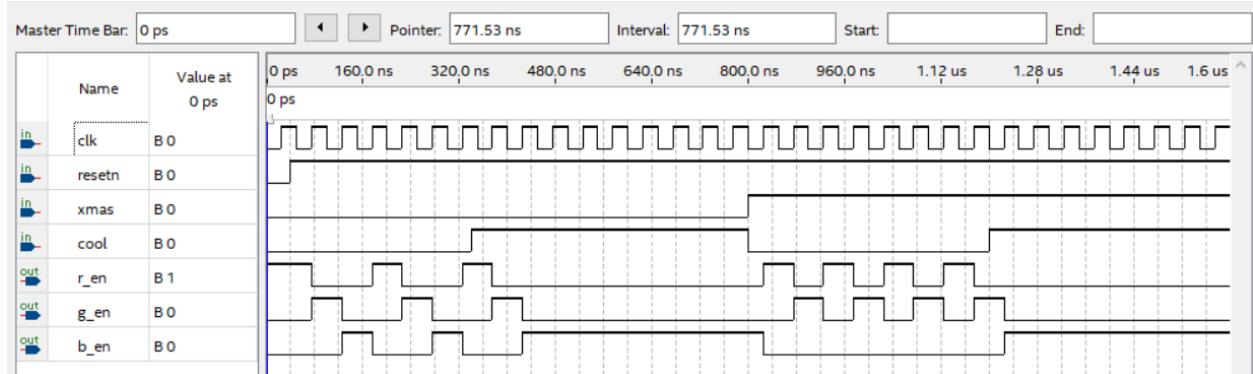


Figure 1. A simulation waveform for all possible input and output combinations of the decorative light system.

```

--StateMachineVHDL.VHD (VHDL)
--This code produces a three-state moore state machine
--Jiacheng Zhang
--ECE2031 L07
--02/13/2022

library ieee;
use ieee.std_logic_1164.all;

entity StateMachineVHDL is
    port(
        xmas, cool, resetn, clk : in std_logic;
        r_en, g_en, b_en       : out std_logic
    );
end entity;

architecture rtl of StateMachineVHDL is
    -- Build an enumerated type for the state machine
    type state_type is (Red, Green, blue);

    -- Register to hold the current state
    signal state : state_type;

begin
    -- Logic to advance to the next state
    process (clk, resetn)
    begin
        if resetn = '0' then
            state <= Red;
        elsif (rising_edge(clk)) then
            case state is
                when Red=>
                    if (xmas = '0') and (cool = '0') then
                        state <= Green;
                    elsif (xmas = '0') and (cool = '1') then
                        state <= Green;
                    elsif (xmas = '1') and (cool = '0') then
                        state <= Green;
                    else
                        state <= Green;
                    end if;
                when Green=>
                    if (xmas = '0') and (cool = '0') then
                        state <= Blue;
                    elsif (xmas = '0') and (cool = '1') then
                        state <= Blue;
                    elsif (xmas = '1') and (cool = '0') then
                        state <= Red;
                    else
                        state <= Blue;
                    end if;
                when Blue=>
                    if (xmas = '0') and (cool = '0') then
                        state <= Red;
                    elsif (xmas = '0') and (cool = '1') then
                        state <= Blue;
                    elsif (xmas = '1') and (cool = '0') then
                        state <= Red;
                    else
                        state <= Blue;
                    end if;
            end case;
        end if;
    end process;

    -- Output depends solely on the current state
    process (state)
    begin
        case state is
            when Red =>
                r_en <= '1';
                g_en <= '0';
                b_en <= '0';
            when Green =>
                r_en <= '0';
                g_en <= '1';
                b_en <= '0';
            when Blue =>
                r_en <= '0';
                g_en <= '0';
                b_en <= '1';
            end case;
        end process;
    end rtl;
end

```

Figure 2. VHDL code used to implement a decorative light system.

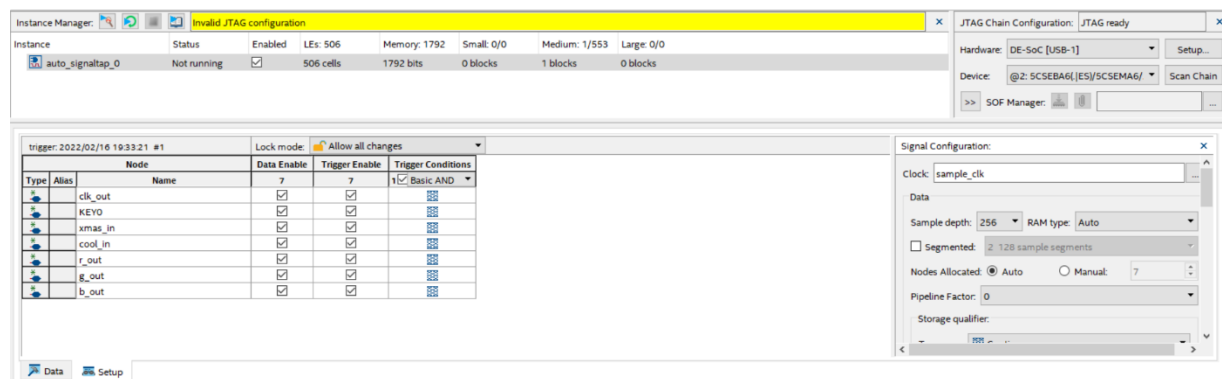


Figure 3. All the necessary signals and acquisition settings of the Signal Tap Logic Analyzer on Quartus.

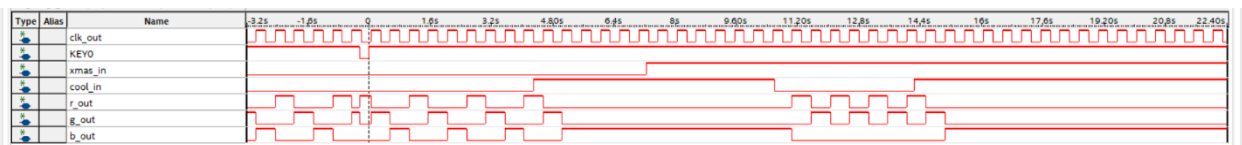


Figure 4. Signal Tap Acquisition to record all possible input and output combinations of the decorative light system with the sample period 0.05 s.