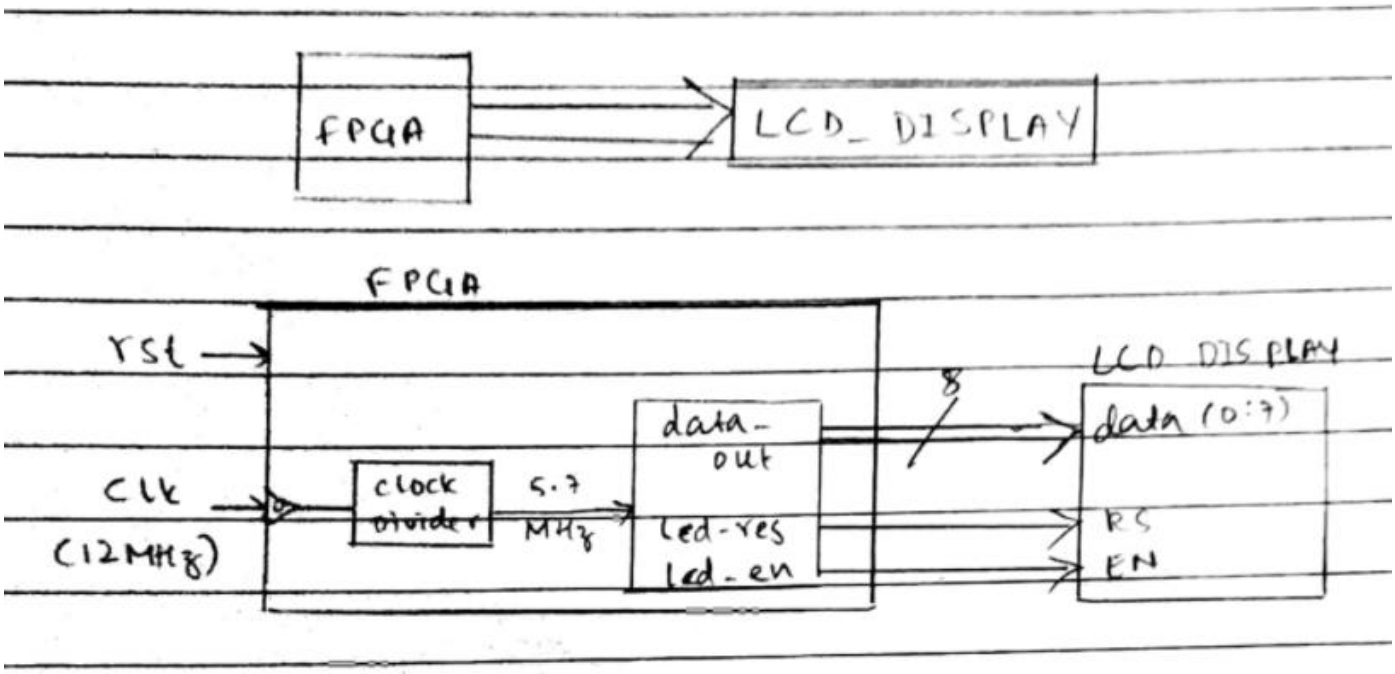
Class	:	BE - 8
Roll No	:	42410
Assignment No	:	4
Assignment Name	:	LCD FPGA Interfacing
Date of Performance	:	17-10-2020

Block Diagram:



Function Table:

PIN ASSIGNMENT (UCF location) for LCD:

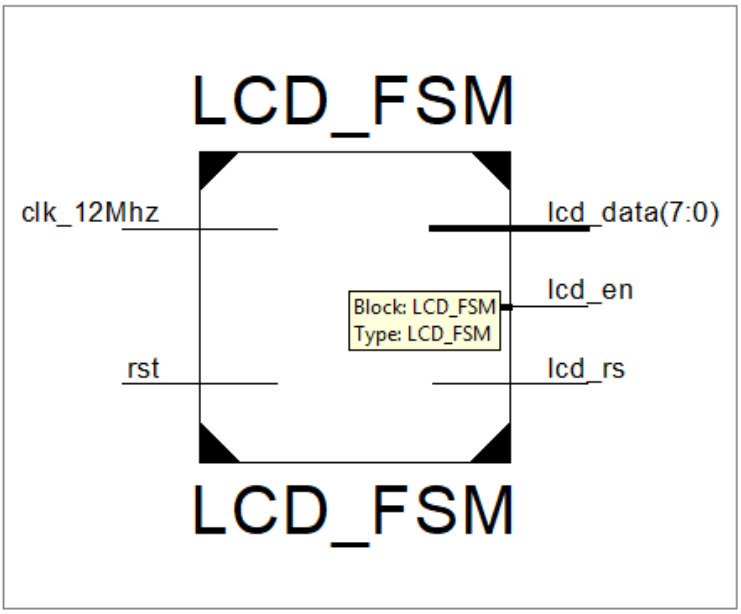
SIGNAL NAME	XC3S250E-PQ208	XC5GLX9-TBG144
LCD-RS	P48	P74
LCD_EN	P49	P72
LCD_DATA0	P47	NC
LCD_DATA1	P41	NC
LCD_DATA2	P39	NC
LCD_DATA3	P35	NC
LCD-DATA4	P33	P66
LCD_DATA5	P31	P61
LCD_DATA6	P29	P59
LCD_DATA7	P24	P58

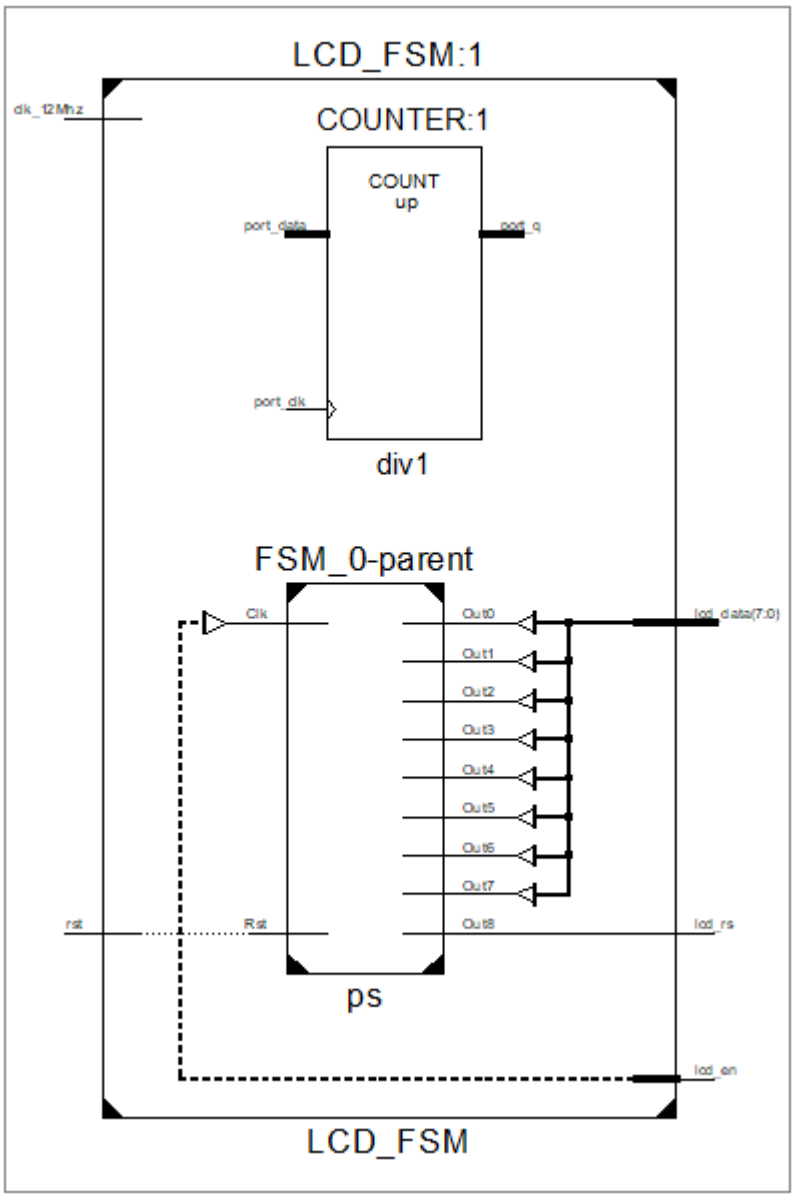
MAIN VHDL PROGRAM

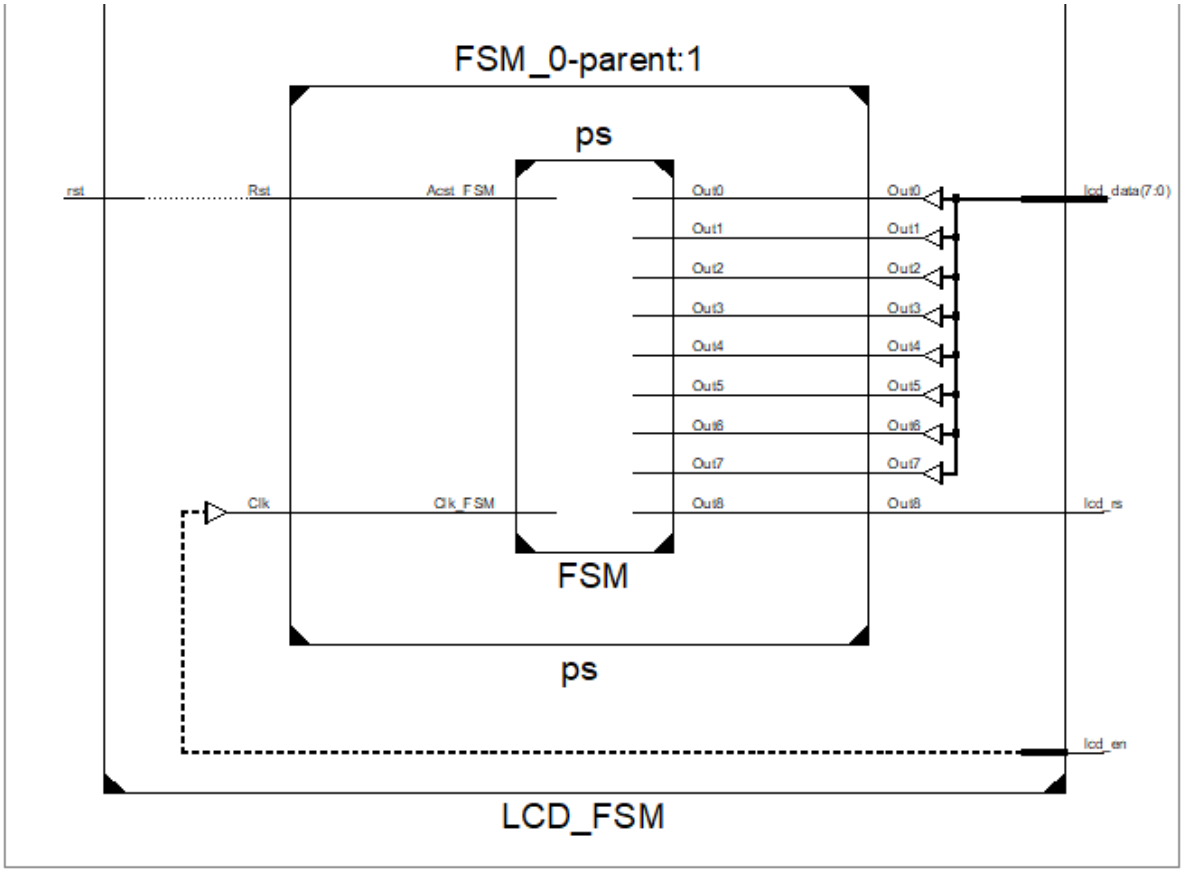
```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity LCD_FSM is
Port (      rst : in std_logic;         -- reset
        clk_12Mhz : in std_logic;          -- high freq. clock
        lcd_rs : out std_logic;        -- LCD RS control
        lcd_en : out std_logic;        -- LCD Enable
        lcd_data : out std_logic_vector(7 downto 0)); -- LCD Data port
end LCD_FSM;

architecture Behavioral of LCD_FSM is
signal div : std_logic_vector(20 downto 0); --- delay timer 1
signal clk_fsm,lcd_rs_s: std_logic;
-- LCD controller FSM states
type state is (reset,func,mode,cur,clear,d0,d1,d2,d3,d4,hold);
signal ps1,nx  : state;
signal dataout_s  : std_logic_vector(7 downto 0); --- internal data command multiplexer
begin

----- clk divider ---------------------------------
process(rst,clk_12Mhz)
begin
if(rst = '1')then
      div <= (others=>'0');
elsif( RISING_EDGE(clk_12Mhz) and clk_12Mhz ='1')then

      div <= div + 1;
      end if;
end process;
-----------------------------------------------------
clk_fsm <= div(20);
----- Presetn state Register ----------------------
process(rst,clk_fsm)
begin
if(rst = '1')then
      ps1  <= reset;
elsif(RISING_EDGE(clk_fsm) and clk_fsm ='1')then
      ps1  <= nx;
end if;
end process;
----- state and output  decoding process
process(ps1)
begin
case(ps1) is
when reset =>
                  nx    <= func;
            lcd_rs_s  <= '0';
                  dataout_s      <= "00111000";           -- 38h

when func      =>
                  nx    <= mode;
                  lcd_rs_s  <= '0';
```

```
                dataout_s      <= "00111000";          -- 38h

when mode   =>
                nx    <= cur;
                lcd_rs_s  <= '0';
                dataout_s      <= "00000110";          -- 06h

when cur=>
                nx    <= clear;
                lcd_rs_s  <= '0';
                dataout_s      <= "00001100";          -- 0Ch  curser at starting point of line1

when clear=>
                nx    <= d0;
                lcd_rs_s  <= '0';
                dataout_s      <= "00000001";          -- 01h

when d0 =>
                lcd_rs_s  <= '1';
                dataout_s      <= "01010000";          -- P
                nx    <= d1;

when d1 =>
                lcd_rs_s  <= '1';
                dataout_s      <= "01001001";          -- I
                nx    <= d2;

when d2 =>
                lcd_rs_s  <= '1';
                dataout_s      <= "01000011";          -- C
                nx    <= d3;

when d3 =>
                lcd_rs_s  <= '1';
                dataout_s      <= "01010100";          -- T
                nx    <= d4;

when d4 =>
                lcd_rs_s  <= '1';
                dataout_s      <= "00100000";          -- space
                nx    <= hold;

when hold     =>
                lcd_rs_s  <= '0';
                dataout_s      <= "00000000";          -- hold
                nx    <= hold;
when others=>
                nx    <= reset;
                lcd_rs_s  <= '0';
                dataout_s      <= "00000001";          -- CLEAR
end case;
end process;
lcd_en <= clk_fsm;
lcd_rs <= lcd_rs_s;
lcd_data <= dataout_s;

end Behavioral;
```

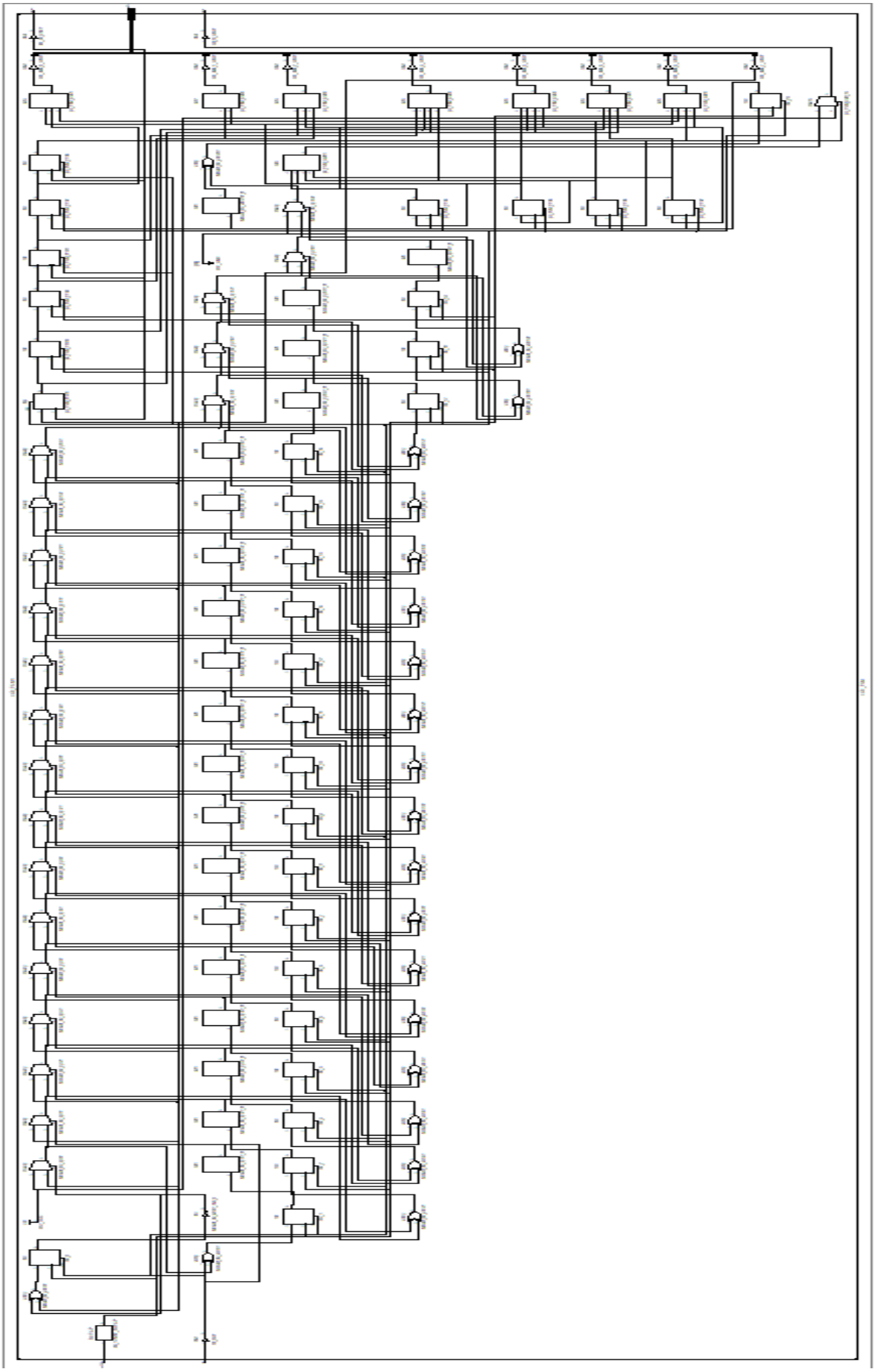
RTL SCHEMATIC

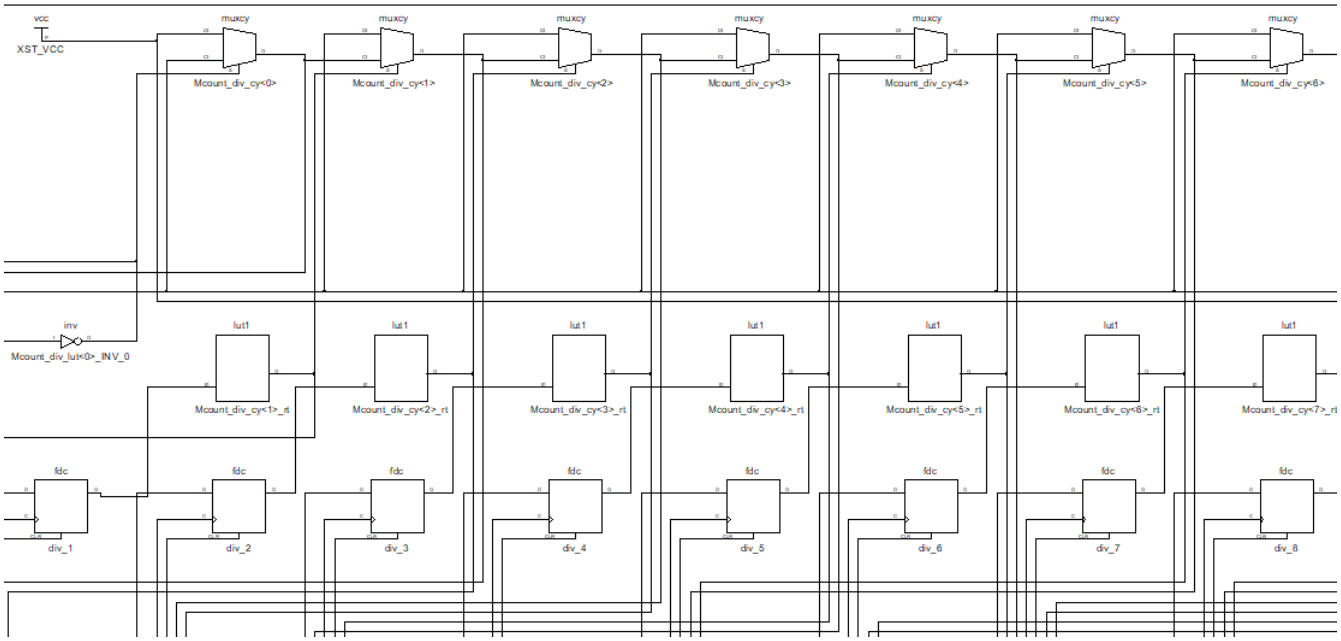





FSM_0-parent:1
ps
rst
Rst
Acst_FSM
Out0
Out1
Out2
Out3
Out4
Out5
Out6
Out7
Out8
lcd_data(7:0)
Clk
Clk_FSM
lcd_rs
FSM
ps
lcd_en
LCD_FSM

TECHNOLOGY SCHEMATIC

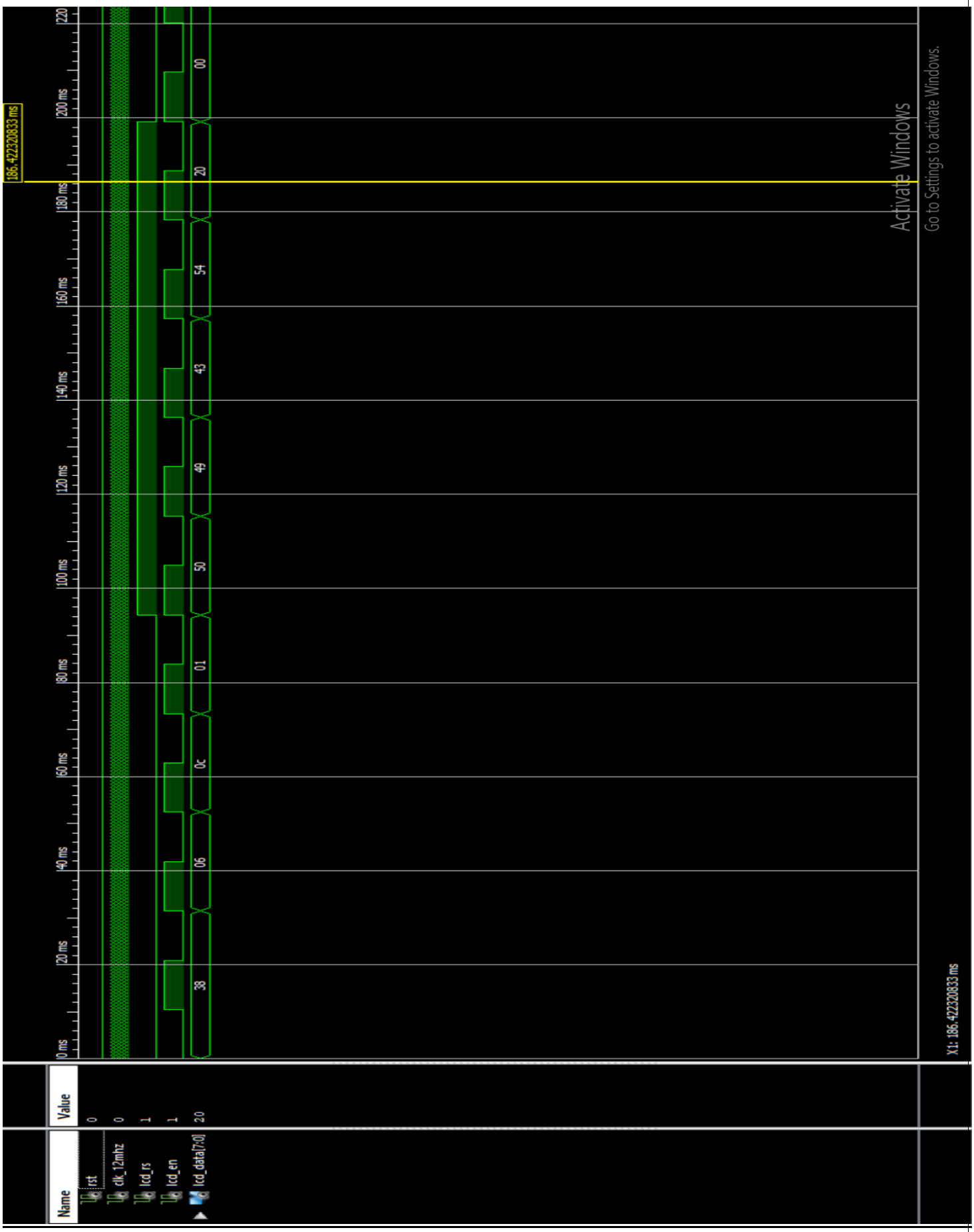



vcc
XST_VCC
muxcy
Mcount_div_cy<0>
muxcy
Mcount_div_cy<1>
muxcy
Mcount_div_cy<2>
muxcy
Mcount_div_cy<3>
muxcy
Mcount_div_cy<4>
muxcy
Mcount_div_cy<5>
muxcy
Mcount_div_cy<6>
inv
Mcount_div_lut<0>_INV_0
lut1
Mcount_div_cy<1>_rt
lut1
Mcount_div_cy<2>_rt
lut1
Mcount_div_cy<3>_rt
lut1
Mcount_div_cy<4>_rt
lut1
Mcount_div_cy<5>_rt
lut1
Mcount_div_cy<6>_rt
lut1
Mcount_div_cy<7>_rt
fdc
div_1
fdc
div_2
fdc
div_3
fdc
div_4
fdc
div_5
fdc
div_6
fdc
div_7
fdc
div_8

TESTBENCH PROGRAM

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
ENTITY LCD_Test IS
END LCD_Test;

ARCHITECTURE behavior OF LCD_Test IS

   -- Component Declaration for the Unit Under Test (UUT)

   COMPONENT LCD_FSM
   PORT(
        rst : IN  std_logic;
        clk_12Mhz : IN  std_logic;
        lcd_rs : OUT  std_logic;
        lcd_en : OUT  std_logic;
        lcd_data : OUT  std_logic_vector(7 downto 0)
       );
   END COMPONENT;
  --Inputs
  signal rst : std_logic := '0';
  signal clk_12Mhz : std_logic := '0';
      --Outputs
  signal lcd_rs : std_logic;
  signal lcd_en : std_logic;
  signal lcd_data : std_logic_vector(7 downto 0);
  -- Clock period definitions
  constant clk_12Mhz_period : time := 10 ns;
BEGIN
       -- Instantiate the Unit Under Test (UUT)
  uut: LCD_FSM PORT MAP (
         rst => rst,
         clk_12Mhz => clk_12Mhz,
         lcd_rs => lcd_rs,
         lcd_en => lcd_en,
         lcd_data => lcd_data
        );
       clk_12Mhz_process :process
  begin
              clk_12Mhz <= '0';
              wait for clk_12Mhz_period/2;
              clk_12Mhz <= '1';
              wait for clk_12Mhz_period/2;
  end process;
  -- Stimulus process
  stim_proc: process
  begin
    rst <= '1';
    wait for 20 ns;
   rst <= '0';
     -- insert stimulus here

     wait;
  end process;

END;
```

ISIM WAVEFORMS:-



PIN-LOCKING REPORT

```
# PlanAhead Generated physical constraints

NET "clk_12Mhz" LOC = P80;
NET "rst" LOC = P204;
NET "lcd_rs" LOC = P48;
NET "lcd_en" LOC = P49;
NET "lcd_data[0]" LOC = P47;
NET "lcd_data[1]" LOC = P41;
NET "lcd_data[2]" LOC = P39;
NET "lcd_data[3]" LOC = P35;
NET "lcd_data[4]" LOC = P33;
NET "lcd_data[5]" LOC = P31;
NET "lcd_data[6]" LOC = P29;
NET "lcd_data[7]" LOC = P24;
```

Conclusion:

Thus we have:

1) Modeled a LCD-FPGA Interfacing using Behavioral Modeling Style.
2) Observed following Schematics: **RTL** & **Technology Schematics** generated **Post-Synthesis**.
3) Interpreted **Device utilization Summary** in terms of LUTs, SLICES, IOBs, Multiplexers & D FFs used out of the available device resources.
4) Interpreted the TIMING Report in terms of Maximum combinational delay as indicative of the Maximum Operating Frequency.
5) Written a TESTBENCH to verify the functionality of LCD-FPGA Interfacing & verified the functionality as per the TRUTH-TABLE, by observing ISIM Waveforms.
6) Used Plan Ahead Editor for pin-locking.
7) Prototyped the FPGA **XC3S250EPQ208-5** to realize LCD-FPGA Interfacing & verified its operation by giving suitable input combination.