IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 1, Issue 5, Oct-Nov, 2013
ISSN: 2320 - 8791
www.ijreat.org

Development of FPGA based PS/2 Mouse and VGA Monitor Interface Technique

P. K. Gaikwad
Department of Electronics
Willingdon College,
Sangli, India

Abstract— **The present paper deals with the development of a Very High Speed Integrated Circuit Hardware Description Language (VHDL) Top level module that makes interfacing between the Personal System, PS/2 mouse and Video Graphics Array (VGA) monitor to locate the position of the mouse and display on the screen in the form of an arrow. The Xilinx Field Programmable Gate Array (FPGA) device was deployed to hardwire the top level entity; developed in this research work. The FPGA board from Digilent Inc., Nexys2; embedded with Spartan 3E FPGA device was deployed for configuring the bit file of the Soft Intellectual Proprietary (IP) Core; constructed using the Structural Modeling Style of the VHDL architecture. The prototype developed here is applicable for position monitoring by attaching the mouse device under the surface-level moving object.**

Keywords— **Position monitoring, PS/2 mouse, VGA, VHDL, Xilinx FPGA**

I. Introduction

The research paper portrays the implementation of Personal System, PS/2 mouse interfacing with a Video Graphics Array (VGA) monitor. The interfacing was technologically advanced using the Field Programmable Gate Array (FPGA). Various Soft Intellectual Proprietary (IP) Cores were instantiated to form a Top level Very High Speed Integrated Circuit Hardware Description Language (VHDL) module to display the current position of mouse device on a VGA monitor. The serial data and serial clock lines were connected to the FPGA module to locate the current position of the mouse. The VGA monitor was interfaced with the FPGA and necessary signals were stimulated to its inputs named as Red, Green and Blue.

The optical mouse is an extremely cost effective displacement sensor. Due to the economics of large volume production, the cost of an optical mouse is extremely low. In several applications, the optical mouse has been demonstrated to be a practicable optical displacement sensor beyond its use as pointing device [1].

The interfacing technique is applicable in detecting the position of a robot moving in two dimensional axes. For that, the mouse would be placed beneath an object to detect the current position of the robot. As reported in [2], the protocol of PS/2 and its interface features of electrical parameters were analyzed. The standard scan code set 2 was also introduced. A real keyboard with PS/2 interface based on embedded system was presented. Both software and hardware implementation were also given.

In this research work, the mouse position was displayed on a VGA monitor in the form of an arrow, and the status of the mouse buttons were also displayed on the Light Emitting Diodes (LEDs) available on the FPGA board. The Xilinx FPGA device Spartan 3E was embedded on the development board: Nexys2; designed and manufactured by Digilent Inc.

II. Soft IP Core Development for PS/2 Mouse and VGA Monitor Interfacing

A VHDL module, named as Mouse Reference Component, as given in [3], is composed of three main modules: the PS/2 interface module, which is responsible for communication between the mouse and the controller; the mouse controller, which receives the data packages from the mouse and outputs the mouse position and button states; and the resolution mouse informer, which provides the active screen boundaries on the X and Y axes. This third component receives the resolution information and calculates the maximum values on the X and Y axes and sets the initial position of the mouse.

The VHDL architecture part (Structural) of the top level entity named as 'MouseDriverTOP' is given below.

```
architecture Structural of MouseDriverTOP is
signal XPOS_int,YPOS_int:std_logic_vector (9 downto 0);
signal BLANK_int      : std_logic;
signal HCOUNT_int,VCOUNT_int: std_logic_vector (10 downto 0);
signal CLK_25MHz_int: std_logic;

component MouseRefComp is
  port ( CLK        : in    std_logic;
         RESOLUTION : in    std_logic;
```

```
        RST       : in    std_logic;
        SWITCH    : in    std_logic;
        LEFT      : out   std_logic;
        MIDDLE    : out   std_logic;
        NEW_EVENT : out   std_logic;
        RIGHT     : out   std_logic;
        XPOS      : out   std_logic_vector (9 downto 0);
        YPOS      : out   std_logic_vector (9 downto 0);
        ZPOS      : out   std_logic_vector (3 downto 0);
        PS2_CLK   : inout std_logic;
        PS2_DATA  : inout std_logic);
end component;

component mouse_displayer is
port (
  clk     : in std_logic;

  pixel_clk: in std_logic;
  xpos    : in std_logic_vector(9 downto 0);
  ypos    : in std_logic_vector(9 downto 0);

  hcount  : in std_logic_vector(10 downto 0);
  vcount  : in std_logic_vector(10 downto 0);
  blank   : in std_logic;

  red_in  : in std_logic_vector(3 downto 0);
  green_in : in std_logic_vector(3 downto 0);
  blue_in  : in std_logic_vector(3 downto 0);

  red_out  : out std_logic_vector(3 downto 0);
  green_out: out std_logic_vector(3 downto 0);
  blue_out : out std_logic_vector(3 downto 0)
);
end component;

component VgaRefComp is
  port ( CLK_25MHz  : in    std_logic;
        CLK_40MHz  : in    std_logic;
        RESOLUTION : in    std_logic;
        RST        : in    std_logic;
        BLANK      : out   std_logic;
        HCOUNT     : out   std_logic_vector (10 downto 0);
        HS         : out   std_logic;
        VCOUNT     : out   std_logic_vector (10 downto 0);
        VS         : out   std_logic);
end component;

begin

Mouse :         MouseRefComp
 port map( CLK                  => CLK_T,
        RESOLUTION => RESOLUTION_T,
        RST        => RST_T,
        SWITCH     => SWITCH_T ,
        LEFT       => LEFT_T ,
        MIDDLE     => MIDDLE_T ,
        NEW_EVENT  => NEW_EVENT_T ,
        RIGHT      => RIGHT_T ,
        XPOS       => XPOS_int ,
        YPOS       => YPOS_int ,
--        ZPOS       => ZPOS ,
        PS2_CLK    => PS2_CLK_T ,
        PS2_DATA   => PS2_DATA_T );

MouseDisplayer :mouse_displayer
port map (clk      =>  CLK_T,
                pixel_clk=>  CLK_25MHz_int,
                xpos     => XPOS_int ,
                ypos     => YPOS_int ,
                hcount   => HCOUNT_int,
                vcount   => VCOUNT_int,
                blank    => BLANK_int ,

                red_in   => red_in_T ,
                green_in => green_in_T,
                blue_in  => blue_in_T ,

                red_out  => red_out_T ,
                green_out=> green_out_T ,
                blue_out =>  blue_out_T);

To_VGA:  VgaRefComp
port map( CLK_25MHz  => CLK_25MHz_int ,
        CLK_40MHz  => '0', --CLK_40MHz  ,
        RESOLUTION => RESOLUTION_T,
        RST        => RST_T,
        BLANK      => BLANK_int , --done
        HCOUNT     => HCOUNT_int,
        HS         => HS_T ,
        VCOUNT     => VCOUNT_int,
        VS         => VS_T );

process (CLK_T,RST_T)
begin
        if RST_T='1' then
                CLK_25MHz_int<='0';
        elsif rising_edge(CLK_T) then
                CLK_25MHz_int<= not CLK_25MHz_int;
        end if;
end process;
end Structural;
```

The Mouse Reference Component consists of global clock input. For that a 50 MHz clock source available on the FPGA board was provided for clocking purpose. An Asynchronous reset signal was also provided to this module from the onboard switch to reset the system at any moment. This Soft IP core given in [3] has provision of selecting the VGA monitor with a resolution of either 640x480 or 800x600 pixels. To select the resolution of the currently connected VGA monitor, there is a special input signal called as 'resolution'. If it is asserted low, then it selects 640x480 resolution monitor, otherwise it enables the 800x600 resolution monitor on assertion of this signal to high. To indicate the status of mouse buttons, i.e. right click, left click or middle press, there are three output signals named as 'left', 'middle', and 'right'. The 'new_event'

signal activates for one clock period, when any mouse movement takes place, newly. There are two signals named as 'PS/2_CLK and 'PS/2_DATA' which communicates with the PS/2 mouse using serial communication protocol. Very importantly to note that, there are two 10 bits signal: 'xpos' and 'ypos'; emerging from this component, which represents the horizontal and vertical position of the mouse. To develop the top level entity and integrate this module, the signals 'xpos' and 'ypos' have to be defined as the internal signals in the architecture body of top level VHDL entity. Because these are necessary to interface with the next reference component called as 'Mouse Displayer Reference Component', and provide the coordinate pertaining to the mouse location.

The Mouse Displayer Reference Component as given in [4] receives the current position of the mouse, the position on screen of the currently displayed pixel, the blank signal, the colour channel data, and the global clock. The cursor image is kept in a 256x2 bit ROM. When the horizontal and vertical coordinates are within the region of the mouse cursor, the non-transparent mouse pixels are displayed.

To move the electron beam of the VGA monitor in the Raster Scanning mode and to enlighten the addressed pixel/(s), it was necessary to provide the pixel address. The horizontal and vertical counts for this purpose were received from another module called as 'VGA Reference Component' as given in [5]. It also generates the horizontal and vertical synchronization pulses to repeat the scanning on new line and new page, respectively. A clock source of 25MHz was provided to this module by dividing the onboard clock source of 50MHz by two. The same clock signal was also given to the 'Mouse Displayer Component', as a pixel clock signal.

Thus, the top level VHDL module was designed using the structural modelling style of the VHDL architecture, by instantiating three components together: Mouse Driver, Mouse Displayer and VGA Component. When the reference components mentioned in [3], [4] and [5] are seen at a glance, it is very difficult for the beginners, to note which input, output (I/O) signals are to be defined as I/O port for the top level module and which are to be declared as internal signals; to interconnect these modules together as such.

III. RTL Synthesis View of the Structured Soft IP Cores

The Fig. 1 shows the Synthesis result with Register Transfer Level (RTL) view. The top entity, 'MouseDriverTOP' shown in Fig. 1 depicts the signals named as 'red_in_T', 'green_in_T' and 'blue_in_T'. These were forced to the values "1111", at the time of entity port declaration, to form display of a common colour for the entire VGA screen. However, the corresponding output colour signals were given to the onboard VGA driver circuit. All the clock signals from different components were tied together and given to the system clock signal 'CLK_T'. In the same way all global reset signals were also connected to form a global and asynchronous reset signal named as 'RST_T'.

The Design window shows that the top level entity 'MouseDriverTOP' integrates three VHDL components: 'MouseRefComp' (its instance name is Mouse), 'mouse_displayer' (instance name is MouseDisplayer) and third one is 'VgaRefComp' (instance name is To_VGA). The design window also shows a User Constraint Format file; named as 'MouseDriverTOP' with an extension of .ucf given to the file.

To interconnect I/O lines of the Xilinx FPGA Spartan 3E device the .ucf file was developed as per the guidelines provided in [6]. After successful completion of the .ucf file, the design went through the process of 'Implementation' and further the bit-file was generated using 'Generate Programming File' option in the Process window of the Xilinx ISE (Version 14.6) Project Navigator. The Adapt Software was used to download this bit-file into the PROM available on the FPGA Nexys2 board.

IV. Hardware Level Verification of the System

The Fig. 2 shows the hardware level interfacing of VGA monitor, PS/2 mouse and the FPGA board. For testing purpose of the system in real time operations, the bit-file was downloaded into the onboard PROM and then the FPGA board was reset. Instantly, the mouse pointer (arrow) was displayed on the VGA monitor and was located at the origin of the screen. The origin of the VGA monitor has negative vertical axis as compared with the conventional method of the plot. However, the horizontal axis is as usual. Due to this, when the FPGA board was reset, the mouse was pointing to the origin of the screen i.e. on left top of the monitor. When the system (not FPGA board) was reset by pressing the BTN0 push button switch on the Nexys2 board, the pointing arrow immediately displayed on the centre of the screen as shown in Fig. 2.

The Mouse Reference Component given in [3] provides the facility of asserting three different signals according to the status of the mouse buttons: right, left and the middle. The output signals associated with these buttons were directly routed towards the output of top level module; hardwired in FPGA system.

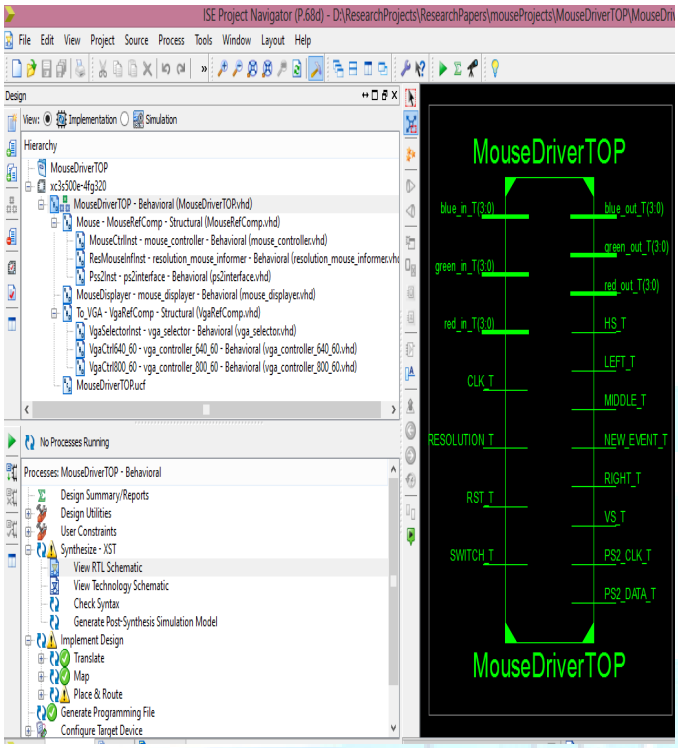

Figure 1. RTL Synthesis view of the Top level Soft IP Core with Xilinx ISE Design Window showing different modules involved; with their hierarchy level.

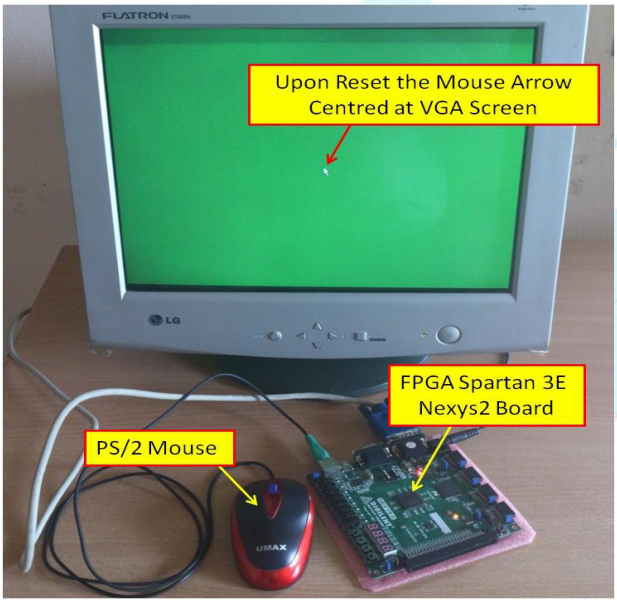

Figure 2. The Complete Hardware Setup to Interface PS/2 Mouse and VGA Monitor with FPGA board

For status visualization of these buttons, the .ucf file was scripted and such output lines were locked with FPGA's I/O pads; where the LEDs are connected on the Nexys2 board.

The Fig. 3 illustrates LEDs switched ON to indicate the status of respective switch of the mouse.

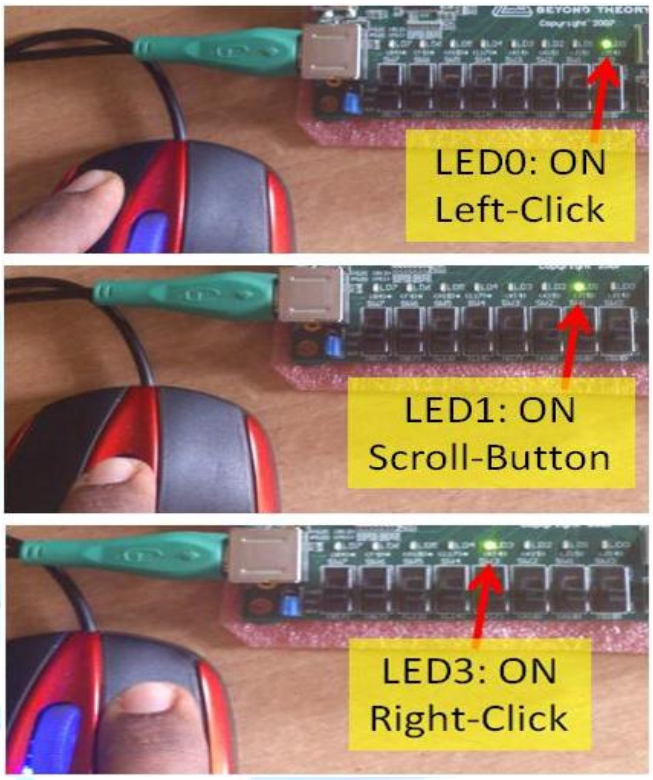

Figure 3. Different LEDs Indicating the Status of the Mouse-switch pressed.

V. Conclusion

The present paper shows the structural modelling style of the VHDL architecture, implemented to integrate different Soft IP Cores and develop a top level module. The PS/2 mouse was interfaced with FPGA Core to communicate using the serial communication protocol; which uses serial data and serial clock signals. To visualize the position of the mouse on the VGA screen, the VGA driver module was integrated with the mouse driver component and a top level VHDL entity was thus formed. There are number of applications of this prototype; for example, to locate the position of a robot by attaching the mouse beneath it.

References

[1] Y. Param Jyothi, E V Narayana, "An Effective Cost Optimization for Thickness Measurement of Metallic Plates", International Journal of Research in Computer and Communication technology, IJRCCT, ISSN 2278-5841, Vol. 2, Issue 1, January 2013, pp. 11-14

[2] CHEN Xi, "Analysis and Application of PS/2 Device Interface Protocol", CNKI Journal, Electronic Design Engineering 2004 -04, retrieved from http://en.cnki.com.cn/Journal_en/I-I135-GWDZ-2004-04.htm

[3] A Guideline Document, "Mouse Reference Component", Digilent Romania, Digilent Inc., September 10, 2008, pp. 1-7, Doc: DSD-0000234

[4] Support Document, "Mouse Displayer Reference Component", retrieved from http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400,789&Prod=NEXYS2

[5] Reference Manual, "VGA Reference Component", Digilent Romania, Digilent Inc., March 28, 2008, pp. 1-4, Doc: DSD-0000241

[6] Board Support Document, "Nexys2 Board Reference Manual", Digilent Inc., 215 E Main Suite D | Pullman, WA 99163, Revision: July 11, 2011, Doc: 502-134.