Design of Image Display Controller Component for VGA Interfaced Monitor using ZedBoard

Divyansh A Thakar[1], Rikin J Nayak[2], Jaiminkumar B Chavda[3], Jay R Patel[4], Mit P Patel[5]

[1, 3, 4] IT Department, [2] E&C Department, [5] CE Department,
Chandubhai S. Patel Institute of Technology, Charotar University of Science & Technology, Changa, Gujarat, India

ABSTRACT

At present, the machines desires to have flexible interfacing and be able to perform massive computation at high data rates to satisfy the need of real-time processing. Field Programmable Gate Arrays (FPGAs) as a programmable computing platform where interfaces and number computation units can be integrated to meet the performance requirements of real-time systems. FPGAs are highly suitable to fulfil these requirements. In most of the consumer display devices, VGA (Video Graphics Array) interface has been still widely accepted as a standard display interface. In this paper we have developed VGA controller on ZedBoard and able to transferred and observed different patter on VGA display.

Keywords : Video Graphics Array, Field Programmable Gate Arrays, VHDL

I. INTRODUCTION

Field Programmable Gate Arrays (FPGAs) are semiconductor, integrated circuits that are designed to be configured by a customer or a designer which has contain configurable blocks of logic along with configurable interconnects between these blocks[1].FPGAs are also known as reconfigurable or reprogrammable devices so, where we can change to new functionality any time. The FPGA contains a new kind of microchips, which combine an FPGA and a processor on a single chip. The biggest advantage of this device is there are possible interfaces between the processor and the FPGA. These types of interfaces are programmable and they provide fast transmission of data [2]. FPGA's configuration is done with the help of hardware description languages (HDL) like Verilog, VHDL. In FPGA there are programmable logic components known as logic elements (LEs). The logic elements can be used to configure complex parallel programming as well as simple logic gates like AND, OR, XOR, NOR, etc. [3]Today's FPGAs can be easily pushed to the 500MHz performance barrier [4].

Nowadays, the research and development of applied digital systems for particular tasks are increasing, such as image processing, ATM machines[5], video games[5], video conference systems [6], face recognition systems [6], surveillance and remote vehicle guidance systems [7], etc. There are many interfaces available in market such as DVI, HDMI, USB, VGA, Camlink, etc. Among them, VGA (Video Graphic Array) is widely used and one of the most popular display interface. By using VGA we can easily connect a system with a display device for displaying images or information. There are many applications such systems like high speed image processing analysis [8], to replace papers used in advertisements[9] etc.

VHSIC Hardware Description Language (VHDL) is highly known and standard hardware description language which is now mostly used by scientists and engineers on digital hardware designs. VHDL is very easy to learn and easy to use and used in Electronic design automation (EDA) and as a parallel programming language [10].VHDL offers many useful features and it is a general-purpose hardware description language.

II. VGA DISPLAY INTERFACE

VGA is an analog connector which deals with high frequency signals. VGA gives maximum resolution 2048x1536 (4:3) at 85 Hz frame rate [11].There are mainly 5 signals required to control VGA display which are red, green, blue, horizontal synchronization, and vertical synchronization. Pixel values need to be continuously on/off at certain frame rate to display image on display. Fig. 1 shows VGA architecture with signals.
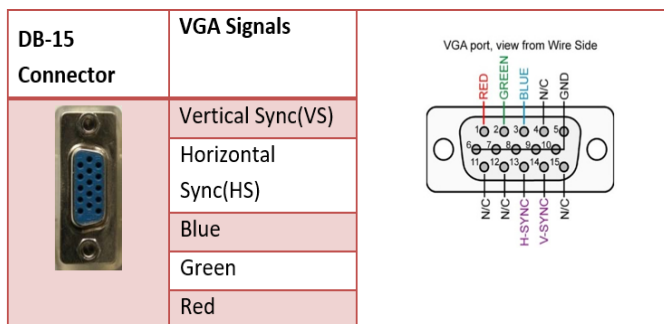

Figure 1: Architecture of VGA

Each line of video starts with an active video region, in which RGB values are given for every single pixel in the line. After active area there will be blanking region, in which black color pixels are transmitted. In the middle of blanking region horizontal sync pulse is transmitted. The blanking interval before the synchronize pulse is known as the "front porch", and the blanking interval after the synchronize pulse is known as the "back porch". Horizontal and vertical synchronization is utilized for taking care of timing of

scan rate [12]. The vertical sync signal is used for indication to the screen to begin displaying a new image, and the screen will begins from the starting pixels that is (0, 0) in left corner and moves to the right side until the point when it achieves the last column. When it achieves the last pixel in the bottom right corner of the screen, it retraces back to the upper left corner and repeats the scanning procedure. Overall, Horizontal Sync performs controlling operation for start and end of line pixel is displaying on the visible area of the monitor. Similarly Vertical Sync performs controlling operation for Start and End of Frame is displaying on the top and bottom visible area of the monitor. VGA need RGB values for each and every pixel. It needs 8 bit value for red, green and blue each so total 24 bits per pixel are required. The VGA uses the red, green and blue colors to generate all other colors.

III. ZYNQ SoC ARCHITECTURE

Xilinx ZYNQ-7000 is a programmable SoC in which there are on chip Artrix-7 FPGA (PL) and Dual core ARM Cortex MP-Core Processor (PS). The Fig. 2 shows the architecture of ZYNQ [13, 14].
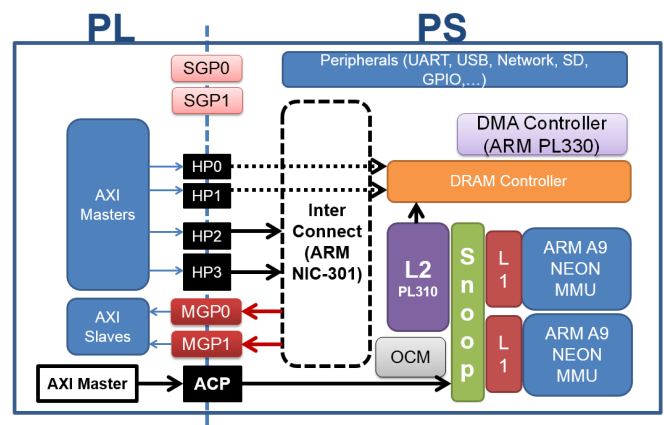

Figure 2: Architecture of ZYNQ SoC

The architecture of ZYNQ SoC is having Processing System (PS), Programmable Logic (PL) and PS-PL Communication interfaces using AXI bus. ZYNQ-PS consists of Interface Controllers for DDR Memory, RS232-UART, SPI, SDIO, Gigabyte Ethernet, USB,

CAN, etc. PS system contains ARM Cortex A9 dual-core processor. Application Processing Unit and system components are connected by AMBA switches. ZYNQ-PL is having Xilinx Kintex-7/Artix-7 FPGA with programmable resources like configurable logic blocks; block RAMs etc. The PL architecture provides user configurable capabilities. Because of availability of PL-PS in single core there are advantages of FPGA like parallel processing due to PL and software/ processing support with of Processor due to PS.

IV. EXPERIMENT AND RESULTS

Fig. 3 shows basic block diagram of the system where we have designed VGA display system using different case studies.
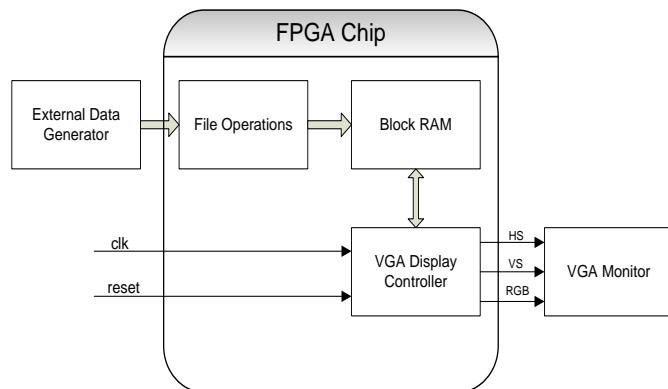

Figure 3: Block diagram of the system

Case 1: Fixed pattern generate from FPGA

In case 1, we have generated fixed pixel values from the FPGA for generating desire pattern on VGA. Fig. 4 shows the result of the VGA display. Here image size is 1024 x 720 and we have transferred the pattern of 256 x 256.

Figure 4: Fixed pattern generate from FPGA

Case 2: Pattern generation from external source.

In case 2, we have generated fixed pixel pattern from one FPGA and transmitted it on 3 wire interface. At the receiver on ZYNQ platform we have designed acquisition system which is acquiring the image and stored it in block memory. After that data from block memory is transmitted on VGA. We have tested this algorithm for both onetime transfer and continuous stream data. Fig. 5 shows the result of the VGA display. Here Image size is 1024 x 720 and we have transferred the pattern of 64 x 64. Here Image size is 64 x64 because FPGA's Block RAM size is fixed and for larger image we can use other external memories like SRAM, SDRAM or DDR memory.

Figure 5: Pattern generation from external source

Case 3: Image display on VGA.

For this case we have developed file related operation which will load the Image of ASCII format in to FPGA and then it will transfer the image on VGA. Fig. 6 shows the results for the same. Here Image size is 1024 x 720 and we have transferred the pattern of 32 x 32. Here also due to limited block memory our resolution of transmitted image is 32 x 32.
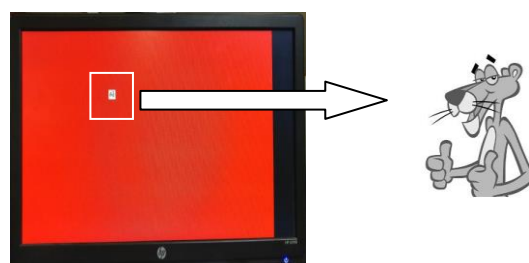

Figure 6: Image display on VGA

The whole program runs on FPGA's inbuilt clock that is 100MHz. To display a picture on display we need to

set pixel values that are for horizontal and vertical counters. VGA uses the hexadecimal values to display each pixel, so we need to convert the whole picture into hexadecimal values with the help of MATLAB and stores it into the text file. The FPGA get access to the pixel values with the help of various file operations and stores it into the block RAM.

V. CONCLUSION

In this proposed system, FPGA is used to develop VGA controller, which can display various patterns and image. The proposed architecture may be used in any FPGA device. In these algorithms, Programming in VHDL makes the design flexible and convenient. The VGA controller is used for image displaying and it is more conventional than other interfaces. Using external source it is possible to observe real time acquire data as image on VGA interface. This can be used for data visualization/analysis in image processing research.

REFERENCES

[1] C. Maxfield, "FPGAs: Instant Access", Elsevier Ltd., USA, 2008.

[2] Stefan Scholl, "The Xilinx Zynq: A Modern System on Chip for Software Defined Radios"

[3] Gopisetty Sri Lakshmi, Rakesh K, "Novel Design and Implementation of VGA Controller on FPGA"

[4] https://www.xilinx.com

[5] NivedithaYadav M, YaseenBasha, Rohith. S, Venkateshkumar.H, "Algorithm to Design VGA Controller on FPGA Board"

[6] Fangqin Ying, XiaoqingFeng"Design and Implementation of VGA Controller Using FPGA"

[7] Seunghun Jin, Dongkyun Kim, ThuyTuong Nguyen, Daijin Kim, Munsang Kim, "Jae WookJeon. Design and Implementation of a Pipelined Datapath for High-Speed Face Detection Using FPGA"

[8] D. Bing, Z. Qi, and W. Rui, "The design and implementation of VGA image controller based on FPGA", College of Automation, Harbin Engineering University, China, 2006.

[9] WoonWoeiKeat, "The design and implementation of VGA controller on FPGA"

[10] Wikipedia, "VHDL",[Online].Available: https://en.wikipedia.org/wiki/VHDL

[11] Altera Corporation, "What is an FPGA?" (26 October 2011), [Online].Available: http://www.altera.com/products/fpga.html

[12] E. Hwang, "Build A VGA Monitor Controller", Circuit Cellar, USA, 2004.

[13] Xilinx Inc.UG585, "Zynq-7000 Extensible Processing Platform Technical Reference Manual", 2012.

[14] Nayak R.J., Chavda J.B. (2018) Comparison of Accelerator Coherency Port (ACP) and High Performance Port (HP) for Data Transfer in DDR Memory Using Xilinx ZYNQ SoC. In: Satapathy S., Joshi A. (eds) Information and Communication Technology for Intelligent Systems (ICTIS 2017) - Volume 1. ICTIS 2017. Smart Innovation, Systems and Technologies, vol 83. Springer, Cham