IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

REVIEW ON FPGA BASED VGA CONTROLLER

Mr. Ashish Kadlag*, Kapaliswaran Pillai, Aswin Pillai and Pratik Thube
Electronics & Tele communications Engg, RMD Sinhgad School Of Engg, India

ABSTRACT

These days devices produced in companies must be highly robust in order to compete with the ever changing demands in products for modern day era. Field-programmable Gate arrays (FPGAs) is best suitable to achieve its basic functioning. FPGAs are efficient, cheap, and portable, according to their implementation specified in hardware description language. Hence, VHDL is best suited in order to accomplish this goal. Programming the gates and counters for FGPA blocks and developing an internal logic, VGA is used. The main purpose of the proposed work is to design and implement VGA Controller on FPGA. VGA controller is designed and VGA controller program is written using VHDL and the corresponding code is executed and implemented on FPGAs chip of Spartan-3A FPGA Development and Educational Board.

KEYWORDS: Field Programmable Gate Arrays (FPGAs), Very High Speed Integrated Circuit Hardware Description Language (VHDL) and Video Graphics Array (VGA).

INTRODUCTION

Field-Programmable Gate Arrays (FPGAs) are digital ICs which contains logic blocks which are configurable along with configurable interconnection between these blocks. These blocks are also known as logic elements (LEs) and an organized system of reconfigurable interconnects that allowing blocks to be connected. Logic elements are be designed to perform tedious combinational functions, or simple logic functions like AND and XOR. In majority of FPGAs, LEs also include memory elements, which can be flip-flops.

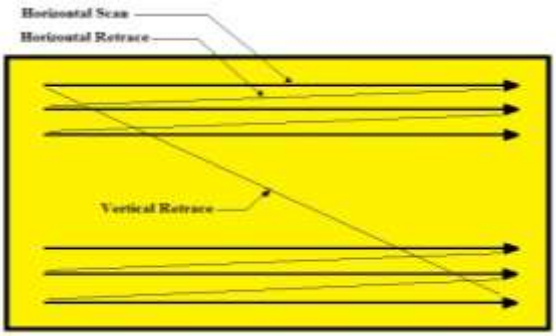
Standard display for video is Video Graphics Array (VGA). For displaying information it gives a simple method so that we could interface a system and a monitor. As a standard display interface, VGA has been widely used. There is much requirement for displaying the result of the process in real time due to the development of embedded system, especially the improvement in image processing with faster rates. Besides, display will be substituting paper in future. Wise words, some things are only possible to accept after witnessing them first hand and picture telling thousand words, presentation can give right data about something. Showcase is utilized when individuals present something. When individuals give presentation, there must be some gadget included to control the display.

VHSIC Hardware Description Language (VHDL) is a prevalent and standard equipment portrayal language which is currently broadly utilized by professionals and researchers on computerized equipment plans. VHDL offers numerous helpful elements for advanced equipment plan, that is, VHDL is a broadly useful equipment describing language that is simple to utilize.The reason for this undertaking is to outline a VGA Controller using VHDL and execute it on FPGA.

LITERATURE SURVEY

The screen for a standard VGA organization contains 640x480 of picture elements called pixels. A picture is shown on the screen by turning on and off exclusively pixels. Turning on one pixel does not speak too much, but joining various pixels creates a picture. The screen constantly looks over the whole screen, quickly turning individual pixels on and off. In spite of the fact that pixels are turned on each one in turn, we get the feeling that every one of the pixels are on since the screen checks so rapidly. This is the reason old screens with moderate sweep rates gleam.

Figure:



Scanning Pattern of VGA Controller

In the figure over the examining procedure begins from row 0, column 0 in the upper left corner of the screen and moves to one side until it achieves the last segment. At the point when the output achieves the end of a column, it remembers to the start of the following line. When it achieves the last pixel in the base right corner of the screen, it backtracks back to the upper left corner and rehashes the checking procedure. Keeping in mind the end goal to decrease glint on the screen, the whole screen must be filtered 60 times each second. This period is known as the revive rate. To decrease glimmer from obstruction from fluorescent lighting sources, revive rates higher than 60 Hz are in some cases utilized as a part of PC screens. Amid the horizontal and the vertical retraces, every one of the pixels are switched off.

The VGA screen is controlled by 5 signals: red, green, blue, horizontal synchronization and vertical synchronization. The three shading signs, all in all alluded to as the RGB signal, control the shade of a pixel at a given area on the screen. They are simple signs with voltages going from 0.7 to 1.0 volt. Varying voltages one can get different intensities of colour. These three-shading signals are dealt with as digital signals, so we can simply switch everyone on or off.

To control timing scan rates horizontal and vertical synchronization signals are utilized. Dissimilar to the three simple RGB signals, these two sync signs are digital signals. In other words, they deal with either logic 0 or logic 1. To control the even deflection circuit in the VGA display the horizontal synchronization sign is utilized, so that the begin and end of a line of pixels is accurately shown over the noticeable showcase region of the screen. Similarly, to control the vertical redirection circuit in the VGA screen the vertical synchronization signal is utilized, so that begin and end of a casing (of lines) is effectively shown between the top and base edges of the unmistakable presentation range of the screen. At the end of the day, horizontal synchronization signal decides the time it takes to output a line, while vertical synchronization signal decides the time it takes to filter the whole screen. By controlling these two sync signals and the three RGB signals,
pictures are shaped on the screen .

To acquire the 640 × 480 screen resolution, a clock with a 25.175 MHz frequency is utilized. A higher clock frequency is required for a higher screen resolution. For the 25.175 MHz clock, the period is as beneath:

$$1/25.175\text{MHz} = 0.0397\ \mu\text{s per clock cycle}$$

In the year 2012 a paper Z. Syed and M. Shaik presented their work on design and implementation of efficient hardware architecture for VGA controllers based on FPGA technology. The design was compatible with PLB bus and had a high potential to be used in Xilinx FPGA-based systems. It had ability to provide multiple display resolutions (upto WXGA 1280×800) and a customizable internal FIFO make the proposed architecture suitable for several FPGA devices. Furthermore, they had also offered a useful software library to enable the text mode feature. These highlight features have been validated through the manifestation of an application.

The same work was carried out by F. Ying and X. Feng from University of Finance and Economics, Hangzhou, China. Their hardware architecture was implemented on Altera EPIC6Q240C8 FPGA（Field Programmable Gate Array） chip. The journal has stated its top layer model project and the timing function simulation. Detailed information was centered on the system structure, hardware design and software programming. That controller was developed using only VHDL supported in the IEEE standards, to ensure the portability with any

manufacturer. The system can display different shade strip, Chinese handwriting and picture. The results show that this intend algorithm fetters useful performance with brief progress period, low resource use, small power consumption and memory usage. Because the data can be sent instantly to monitors, the design can quicken data processing, rectify system reliability in real time and protect hardware resource.

In the same year, the same work was carried out by Radi, Caleb, Zainudin and Ismail from Technical University of Malaysia Malacca.
In order to design and instrument VGA Controller on FPGA, Verilog HDL was used. Verilog HDL was used to describe and program the gates and counters in FPGA blocks in order to construct an internal logic circuit in FPGA. The main purpose of that work was to design and implement VGA Controller on FPGA. Hence, the outline for VGA Controller was designed and the VGA Controller program is written based on the block diagram using Verilog HDL. Also, functions need for VGA Controller are confined in the Verilog code and test bench was created to test the functions written to ensure the FPGA VGA Controller works correctly and accurately without errors. Finally, the completed program was implemented on FPGA of Altera DE2-115 Development and Educational Board.

In year 2015 R. Wasu and V. Wadhankar worked on implementation of the same. They developed controller using Verilog HDL supported in the IEEE standards, to ensure the portability with any user. The system can show any picture. The results show that this intended algorithm gives excellent performance with brief progress period, small power consumption and memory usage.

Similar work was carried out by S. Ajith, S. Bandarupalli and M. Borgaonkar. Using FPGA as its core, they designed an image processing solution in hardware. Their aim was to retrieve a picture from source, discover the presence of an object based on colour and compute its parameters like region and centroid, while displaying the picture on a VGA screen. Their detail outlines the implementation of the performance in two phases. First phase narrates the process of exhibiting a picture on a VGA monitor using SRAM as the video memory. The second phase depicts the implementation of a blob discovery algorithm supported on sequential joined component labelling algorithm. A rework conversion of the labelling algorithm is fulfilled, which enables the processing of an whole picture in a single pass through the picture. This work enables the processing action to be inserted in the pixel data path from the picture source to the video memory.

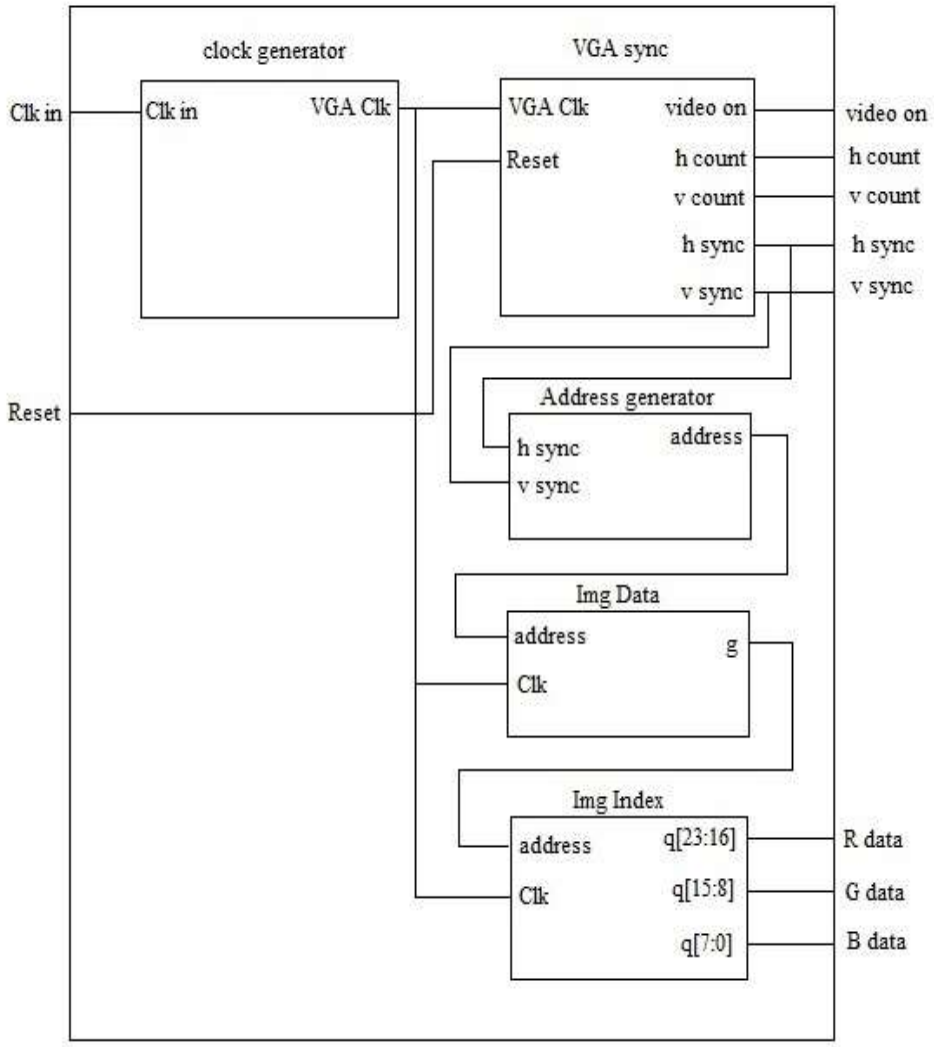
OUTLINE OF VGA CONTROLLER

A. Block Diagram

Alluded to figure, the work of "clock generator" block is to down convert the recurrence of data clock. In the interim, "vga_sync" piece is utilized to produce timing and synchronization signals. The "h_count" and "v_count" demonstrate the relative positions of the outputs and basically indicate the present's area pixel while the "h_sync" sign determines the obliged time to sweep a line, and the "v_sync" sign indicates the obliged time to examine the whole screen. "vga_sync" block likewise creates the "video_on" signal which demonstrates whether to show or hide the monitor screen. Besides that, "address generator" block is utilized to produce address for the "img_data" obstruct by utilizing the "h_sync" and "v_sync" signal. "img_data" block will get the record information (q) from the MIF record as per the location created.

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

Figure:



Block diagram of VGA Controller

Note that the record information are associated with the "img_index" block to use as the location. The "img_index" block will get the RGB information (q) from MIF document as indicated by the location created (record information). The RGB information comprise of 24-bits, though "q [23:16] ", "q [15:8]" and "q [7:0]" demonstrate the "R_data", "G_data" and "B_data" individually.

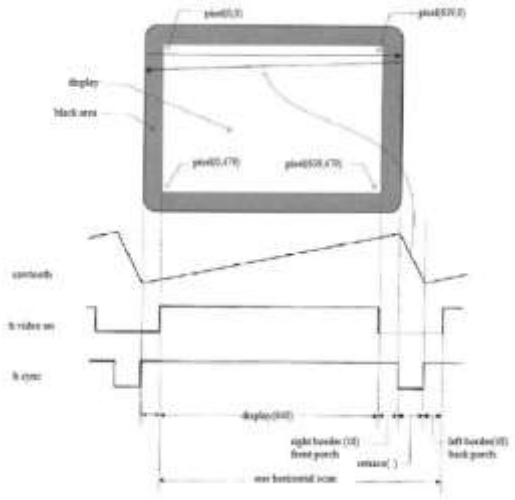
B. Design Flow of VGA Synchronization

The feature synchronization circuit produces the hsync signal, which determines the obliged time to cross (output) a line, and the vsync signal, which indicates the obliged time to navigate (filter) the whole screen. Ensuing talks depend on a 640-by-480 VGA screen with a 25-MHz pixel rate, which implies that 25M pixels are handled in a moment. Note that this determination is otherwise called the VGA mode. The screen of a CRT screen for the most part incorporates a little dark outskirt, as demonstrated at the highest point of Figure. The centre rectangle is the noticeable segment. Note that the vertical's direction hub increments descending. The top's directions left and base right corners are (0, 0) and (639,479), separately.

Horizontal synchronization

A detailed timing chart of one horizontal output is indicated in figure.

Figure:



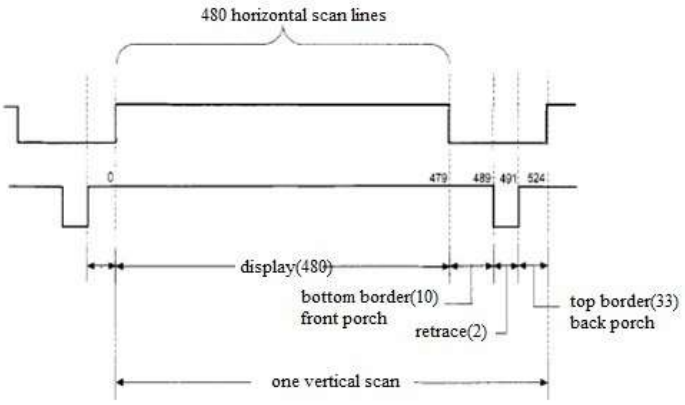
Horizontal synchronization signal-timing diagram

A time of the hsync sign contains 800 pixels and can be isolated into four regions:
Display: Region where the pixels are really shown on the monitor. The distance of this region is 640 pixels.
Retrace: Region in which the electron rays revert to the left margin. The video signal should be lamed (i.e., dark), and the duration of this region is 96 pixels.
Right edge: Region that configures the right edge of the display region. Also called as the *front porch*. The video signal should be lamed, and the duration of this region is 16 pixels.
Left border: Region that forms the left border of the display region. Also known as the *back porch.*

The hsync signal can be acquired by an uncommon mod-800 counter and an interpreting circuit. The numbers are checked on the highest point of the hsync signal in Figure. We purposefully begin the checking from the earliest starting point of the display region. This permits us to utilize the counter yield as the horizontal (x-hub) coordinate. This yield constitutes the pixel-x signal. The hsync sign goes low when the counter's yield is somewhere around 656 and 751.

Vertical synchronization
Amid the vertical output, the electron beams move step by step from start to finish and afterward come back to the top. This compares to the time needed to revive the whole screen. The organization of the vsync sign is like that of the hsync signal, as demonstrated in Figure 5. The time unit of the development is spoken to as far as horizontal sweep lines.

Figure:



Vertical synchronization signal-timing diagram

A time of the vsync sign is 525 lines and can be separated into four regions:
Display: Region where the pixels are really shown on the monitor. The distance of this region is 480 pixels.
Retrace: Region in which the electron rays revert to the top left corner.
Top border: Region that forms the top edge of the display region. Also known as *top porch.*
Bottom border: Region that forms the bottom edge of the display region. Also known as *bottom porch*.
The vsync signal can be acquired by an uncommon mod-525 counter and an interpreting circuit. The numbers are checked on the highest point of the vsync signal in figure. We purposefully begin the checking from the earliest starting point of the display. This permits us to utilize the counter yield as the vertical (y-hub) coordinate. This yield constitutes the pixel-y signal. The vsync sign goes low when the counter's yield is somewhere around 489 and 491.

C. Outline Flow of VGA Controller
Most importantly, "vga_clk" is created from a input clock. At that point, "reset" is created for "vga_sync" module. The timing outline for horizontal and vertical sweep is produced too. After that, address is produced for the "img_data" module, by utilizing the "h_sync" and "v_sync" signal from "vga_sync" module. "img_data" module will get the record information (q) from the MIF document as per the location produced. Note that the record information is associated with the "img_index" module to use as the location. In this way, "img_index" module will get the RGB_data_raw (q) from MIF record as indicated by the location created (file information). Since the RGB_data_raw comprise of 24-bits, it is isolated into "q [23:16] ", "q [15:8]"and "q [7:0]", which demonstrates the "R_data", "G_data" and "B_data" separately. Next, the associations with the yield port are made and picture is shown on display screen.

D. Understanding Interfacing with VGA
The Spartan 3 FPGA board that we used for this project has a built-in VGA port with five active signals as, hsync, vsync, and video signals - red, green, blue. The video signal for VGA is an analog signal, and so a typical video controller uses a D-A converter. However, in the S3 FPGA, only 1 bit is used per color. So it does not require a D-A converter. There are three video color signals available, so we can have eight different colors which can be displayed on screen. For that we need to give proper binary input combinations to VGA port. Following table shows the different possible color combinations:

Tables:

Table 1 Three Bit VGA Combination

Red(R)	Green(G)	Blue(B)	Resulting Color
0	0	0	Black
0	0	1	Blue
0	1	0	Green
0	1	1	Cyan
1	0	0	Red
1	0	1	Magenta
1	1	0	Yellow
1	1	1	White

CONCLUSION
All in all, Field-Programmable Gate Array (FPGA) is better innovation to be utilized as a part of adding to a VGA Controller. By utilizing VHSIC Hardware Description Language (VHDL) on FPGA, VGA Controller could be built effortlessly without building the circuit physically; just to compose a behavioural model or couple of behavioural models in view of its rationale streams, then reproduce it with test seats, incorporate it with netlist, lastly program it onto FPGA. It is extremely powerful as this VGA controller just needs new information to change to other outline show. In this way, FPGA-based VGA controller may be a decent decision as it is anything but difficult to be composed and modest to be utilized.

ACKNOWLEDGEMENTS
We thank our colleagues from RMD Sinhgad School of Engineering who provided insight and expertise that greatly assisted the research, although they may not agree with all of the interpretations of this paper.

We thank Mr.Ashish Kadlag for assistance with VLSI programming and for comments that greatly improved the manuscript.We would also like to show our gratitude to the Prof. Rajesh Shekokar for sharing their pearls of wisdom with us during the course of this research.

We are also immensely grateful to people who we have referred for their comments on an earlier version of the manuscript, although any errors are our own and should not tarnish the reputations of these esteemed persons.

REFERENCES

[1] Zaheerudin Syed, Munwar Shaik, Kaktiya Institute of Technology & Science,Warangal India, "FPGA Implementation of VGA Controller", in Research Gate publication,January 2012.
[2] FangquinYing, Xiaoqing Feng,College of Dong Fang, Zhejang University of Finances and Economics,China, "Design and Implementation of VGA Controller Using FPGA",in International Journal of Advancements in Computing Technology(IJACT) Volume4, Number17,September. 2012
[3] Radi H.R., Caleb W.W.K., M.N. Shah Zainuddin, M. Muazfar Ismail, Universiti Teknikal Malaysia Melaka,"The Design and Implementation of VGA Controller on FPGA", in International Journal of Electrical & Computer Sciences IJECS-IJENS Vol:12 No:05, October, 2012.
[4] Renuka Wasu, Vijay R. Wadhankar, Agnihotri College of Engineering, Wardha, India, "Review Design of VGA Controller Using FPGA", **in** International Journal of Science and Research (IJSR), Volume 4 Issue 2, February 2015.
[5] S. Ajith, S. Bandarupalli, M. Borgaonkar, "Image Processing Using FPGA", ECE project report.

AUTHOR BIBLIOGRAPHY

	**Mr. Ashish Kadlag** Asst.Prof. E&Tc, RMD Sinhgad School Of Engg.,Savitribai Phule Pune University, Pune
	**Kapaliswaran Pillai** B.E.E&Tc, RMD Sinhgad School Of Engg.,Savitribai Phule Pune University, Pune
	**Aswin Pillai** B.E.E&Tc, RMD Sinhgad School Of Engg. ,Savitribai Phule Pune University, Pune
	**Pratik Thube** B.E.E&Tc, RMD Sinhgad School Of Engg.,Savitribai Phule Pune University, Pune