"I admit to have read this report and it has followed the scope and quality in partial fulfillment of requirements for The Bachelor Degree of Electronic Engineering (Computer Engineering)"

Signature : .....

Supervisor Name : En Nor Zaidi Bin Haron

Date : 30/4/07

© Universiti Teknikal Malaysia Melaka

VIDEO GRAPHIC ARRAY (VGA) GENERATOR USING COMPLEX PROGMMABLE LOGIC DEVICE (CPLD)

ADY IZWAN BIN OMAR

This Report Is Submitted In Partial Fulfillment of Requirements for the Bachelor Degree of Electronic Engineering (Computer Engineering)

Faculty of Electronics Engineering and Computer Engineering
Universiti Teknikal Malaysia Melaka

APRIL 2007

© Universiti Teknikal Malaysia Melaka

UNIVERSTI TEKNIKAL MALAYSIA MELAKA
FAKULTI KEJURUTERAAN ELEKTRONIK DAN KEJURUTERAAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek	:	Video Graphic Array (VGA) Generator Using Complex Programmable Logic Device (CPLD)
Sesi Pengajian	:	2006 / 2007

Saya ADY IZWAN BIN OMAR
(HURUF BESAR)

mengaku membenarkan Laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (√) :

☐	**SULIT***	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	**TERHAD***	(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
☑	**TIDAK TERHAD**	

Disahkan oleh:

(TANDATANGAN PENULIS)

Alamat Tetap: 8 Jalan 48 Desa Jaya,

Kepong 52100

Kuala Lumpur

Tarikh: 30/4/2007

(COP DAN TANDATANGAN PENYELIA)

NOR ZAIDI B HARON
Pensyarah
Fakulti Kej Elektronik dan Kej Komputer (FKEKK),
Universiti Teknikal Malaysia Melaka (UTeM),
Karung Berkunci 1200,
Ayer Keroh, 75450 Melaka

Tarikh: 30/4/07

© Universiti Teknikal Malaysia Melaka

Dedicated to:

Ayah, Ibu, my family, Mr. Nor Zaidi Bin Haron, Ms. Shafinaz Binti Saleh and all my friends that have helped me during the process of completing this project. Thanks for all the support.

© Universiti Teknikal Malaysia Melaka

ACKNOWLEDGEMENT

First of all, I would like to thank Allah the All Mighty, which with his bless, I manage to complete this project and thesis.

I would like to express my greatest gratitude and thanks to my supervisor, Mr. Nor Zaidi Bin Haron, for accepting me as his project student and for his valuable ideas, advice and help in the supervision and discussions of this Final Year Project. In fact, he gave me guidance when obstacles arise throughout this period of time. Once again, I thank him for his tolerance and endeavors.

I am especially grateful to my parent and members of my family, for all their support and understanding along my study. Lastly, my grateful goes to all my colleagues who give me guidance and help in completing this project.

© Universiti Teknikal Malaysia Melaka

ABSTRACT

This project is to design, simulate, synthesize a VGA generator written in Very High Speed Integrated Circuit Hardware Design Language (VHDL) code and implement into Complex Programmable Logic Device (CPLD). This module eliminates the use of CPU to display or textual data on VGA monitor. The design will be integrated with Programmable Logic device-based Logic Analyzer to have a complete system

© Universiti Teknikal Malaysia Melaka

ABSTRAK

Projek ini adalah untuk merekacipta, simulasi, '*synthesize*' penjana VGA yang ditulis di dalam *Very High Speed Integrated Circuit Hardware Design Language (VHDL)* dan dilaksanakan ke atas *Complex Programmable Logic Device (CPLD)*. Modul ini adalah bertujuan untuk mengelakkan penggunaan CPU untuk menghasilkan paparan atau text data ke monitor.. Rekaan ini akan digabungkan dengan *Programmable Logic device-based Logic Analyzer* untuk menghasil satu projek yang lengkap.

© Universiti Teknikal Malaysia Melaka

CONTENTS

© Universiti Teknikal Malaysia Melaka

© Universiti Teknikal Malaysia Melaka

LIST OF TABLES

© Universiti Teknikal Malaysia Melaka

LIST OF FIGURES

© Universiti Teknikal Malaysia Melaka

© Universiti Teknikal Malaysia Melaka

LIST OF ABBREVIATIONS

VGA	-	Video Graphic Array
CPU	-	Central Processing Unit
VHDL	-	Very High Speed Integrated Circuit Hardware Description Language
HS	-	Horizontal
VS	-	Vertical
RGB	-	Red Green Blue
TTL	-	Transistor Transistor Logic
CRT	-	Cathode Ray Tube
IEEE	-	Institute of Electrical and Electronics Engineers
DOS	-	Disk Operating System
ROM	-	Read Only Memory
RAM	-	Random Access Memory
PAL	-	Programmable Array Logic
PLA	-	Programmable Logic Array
RTL	-	Register Transfer Level
IC	-	Integrated Circuit
CPLD	-	Complex Programmable Logic Device
HDL	-	Hardware Description Language
FPGA	-	Field Programmable Gate Array
FPLD	-	Field Programmable Logic Device
PC	-	Personal Computer
PCB	-	Printed Circuit Board
ADC	-	Analog to Digital Converter
DC	-	Direct Current

© Universiti Teknikal Malaysia Melaka

CHAPTER 1

1.0 INTRODUCTION

Chapter 1 starts with the background of the project. It is followed by objectives and scope of the project. The overview method of project is presented in fourth part and lastly summary of the thesis is described.

1.1 OBJECTIVES PROJECT

The objective of this project is to generate textual and graphical data on VGA monitor. Data from Logic Analyzer designed by other student will be combined with this design. The final aim is to have single chip PLD-based Logic Analyzer that display the output measurement to monitor without interfaced to CPU.

1.2 BACKGROUND OF PROJECT

Nowadays, most of the logic analyzer uses C.P.U. to display the output on the monitor. The problem of existing logic analyzer is that it requires software in order to function properly. Furthermore, it is not portable. The objective of this module is to eliminate the use of C.P.U. to display graphical or textual data on VGA monitor.

© Universiti Teknikal Malaysia Melaka

1.3 SCOPE OF WORK

This final year project is basically the application of computer engineering. The project is divided into two parts; digital logic design and hardware fabrication. The design will be implementing in CPLD and then fabricated on PCB. This project will integrated with programmable logic device-based Logic Analyzer to have a complete system.

Xilinx ISE webpack v8,1i will be used to write CPLD code for the digital logic design; VGA generator. The design is then simulated using ModelSim XE III 6.0d before downloaded into CPLD.

The programmed CPLD will be fabricated on PCB. Protel DXP will be used to draw the PCB. Testing will be carried out to test its functionality.

1.4 FLOW OF THESIS

This thesis is divided into five (5) chapters. Introduction that consists of objective, scope and project methodology is described in Chapter 1. Literature studies are described in Chapter 2, on the research paper on Digital Oscilloscope, PC-Based Logic Analyzer, VGA Display and VHSIC Hardware Description Language (VHDL). The project methodology is described in Chapter 3. There are two part that is the software and hardware part in building this project and this chapter will be describing about the technique and method used. This chapter will consist of the block diagram, the circuit of the project and flow chart that will describe the flow of the project. The result and analysis is described in Chapter 4. The result of the project, the analysis of the project that have been achieve, the circuit that have been build and the result of simulation and testing

© Universiti Teknikal Malaysia Melaka

of the project will be describe in this chapter. The last chapter in this thesis is conclusion and suggestion. In this chapter, all the achievement and studies that have been made in order to understand and being able to complete this project will be summaries. Other than that, suggestion for future improvement of the project will be describe.

© Universiti Teknikal Malaysia Melaka

CHAPTER 2

2.0 LITERATURE STUDIES

2.1 INTRODUCTION

In this chapter, details of explaination about the theory that have bee used during the making of this project will be discussed. To achieve the objective, research have been done about the Video Graphic Array (VGA).

2.2 DIGITAL OSCILLOSCOPE

In [1], a digital oscilloscope is designed. The block diagram in figure 2.1, illustrates two entities: **vgamemory** and **vga**.

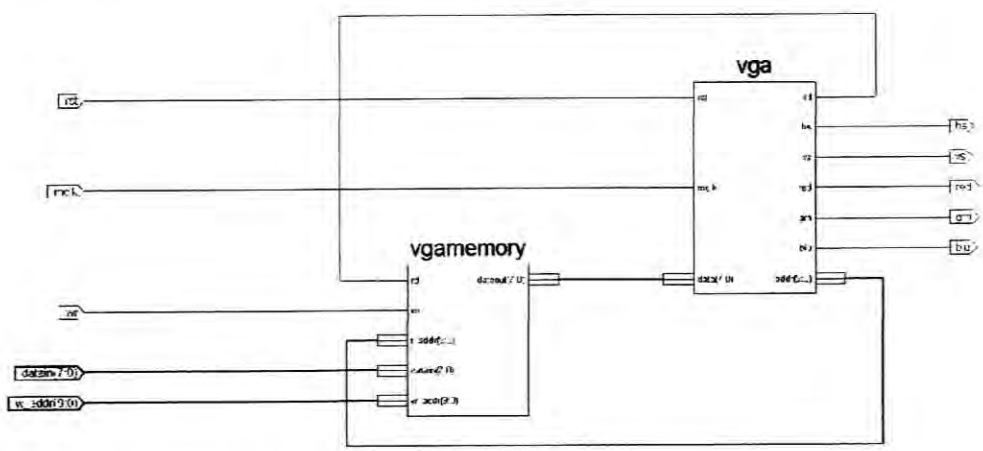

Figure 2.1: Block diagram of VGA Generator [1]

© Universiti Teknikal Malaysia Melaka

The **vgamemory** entity represents a dual-port memory module where the values obtained from the A/D conversion are buffered prior to being shown on the screen. The capacity of the memory array is 640 bytes. Each entry in the memory corresponds to a column on the screen. The value contained in the respective entry denotes the vertical coordinate of the pixel which will be lit up in the corresponding column; e.g. if the value from index 20 in the memory array is 42, this means that the pixel found at the intersection of column #20 and row #42 will be lit.

The contents of **vgamemory** are written by the acquisition module. This operation is not synchronous; rather it takes place on the falling edge of the *wr* signal. The contents of the memory are read by the vga controller, whose description follows.

The **vga** entity (implemented in vga.vhd) is a video controller which drives the video signals (*hs*, *vs*, *red*, *grn*, *blu*) needed by an external monitor. Based on these signals the monitor can create an image on the display.

The implementation of **vga** extends an already existing video controller module, originally created for Digilent Pegasus board. It adds signals for the logical coordinates of spot on the screen (x and y) and rearranges the *red*, *grn* and *blu* signals, such that they display a different image. The logical coordinates are obtained by subtracting from the horizontal (*hc*) and vertical (*vc*) counters the horizontal (*hbp*) and vertical (*vbp*) back porches respectively. The horizontal logical coordinate (x) denotes a column on the screen. Its value is routed out the controller through the *addr* signal and is used to address the video memory. The value obtained from the memory (*data*) represents the logical vertical coordinate which denotes a row on the screen. Hence, the plot of the signal on the screen will consist of lit pixels having coordinates equal to corresponding [x, *data*] pairs.

When the oscilloscope is in hold mode, the video memory is not refreshed. This freezes the plot of the signal on the monitor, because the same video memory content is displayed over and over again.

© Universiti Teknikal Malaysia Melaka

Compared to vga generator, there are quite similar because the concept of this project is similar but the digital oscilloscope has an extra feature. The digital oscilloscope has a hold button to display the waveform.

2.3 PC-BASED LOGIC ANALYZER

In [2], a PC-Based Logic Analyzer is designed. The purpose of this project is to create a useful yet easy to build Logic Analyzer that would provide much of the functionality of a real Logic Analyzer. The biggest drawback that the computer has is its sampling rate. The program was written on a 75 MHz Pentium computer which provided accurate sampling rates up to 10 kHz. Although this doesn't match the speed of real Logic Analyzers, this should be sufficient speed to be useful for many projects. Figure 2.2 shows the process flow of PC-Based Logic Analyzer.

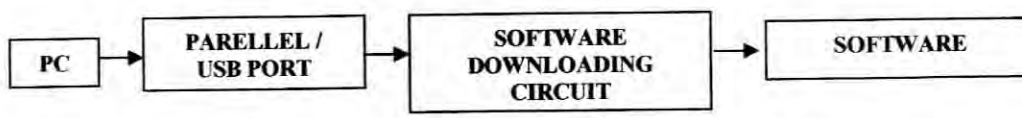

Figure 2.2: The process flow of PC-Based Logic Analyzer [2]

The only requirements for the computer are a VGA display and a parallel port. There is no speed requirement for the computer but of course the higher the clock rate of the computer the higher the obtainable sampling rate will be. Also, if you are using Windows 95, the program must be run in DOS mode.

This project also using the parallel port as the interface to connect to the CPU. Besides that, using the software to display the result to monitor.

© Universiti Teknikal Malaysia Melaka

2.4 VGA DISPLAY GENERATION

In [3], to understand how it is possible to generate a video image, it is first necessary to understand the various components of a video signal. A VGA video signal contains 5 active signals. Two signals compatible with TTL logic levels, horizontal sync and vertical sync, are used for synchronization of the video. Three analog signals with 0.7 to 1.0-Volt peak-to-peak levels are used to control the color. The color signals are Red, Green, and Blue. They are often collectively referred to as the RGB signals. By changing the analog levels of the three RGB signals all other colors are produced.

The technology used to display a video image dictates the very nature of the video signals. The major component inside a VGA computer monitor is the color CRT or Cathode Ray Tube. The electron beam must be scanned over the viewing screen in a sequence of horizontal lines to generate an image. The deflection yoke uses magnetic or electrostatic fields to deflect the electron beam to the appropriate position on the face of the CRT. The RGB color information in the video signal is used to control the strength of the electron beam. Light is generated when the beam is turned on by a video signal and it strikes a color phosphor dot or line on the face of the CRT. The face of a color CRT contains three different phosphors. One type of phosphor is used for each of the primary colors of red, green, and blue. In standard VGA format, as seen in Figure 9.2, the screen contains 640 by 480 picture elements or pixels. The video signal must redraw the entire screen 60 times per second to provide for motion in the image and to reduce flicker. This period is called the refresh rate. The human eye can detect flicker at refresh rates less than 30 Hz. To reduce flicker from interference from fluorescent lighting sources, refresh rates higher than 60 Hz are sometimes used in PC monitors. The onboard clock on the Altera UP 1 board produces a fixed 60Hz refresh rate. The color of each pixel is determined by the value of the RGB signals when the signal scans across each pixel. In 640 by 480-pixel mode,

© Universiti Teknikal Malaysia Melaka

with a 60Hz refresh rate, this is approximately 40 ns per pixel. A 25 MHz clock has a period of 40 ns.

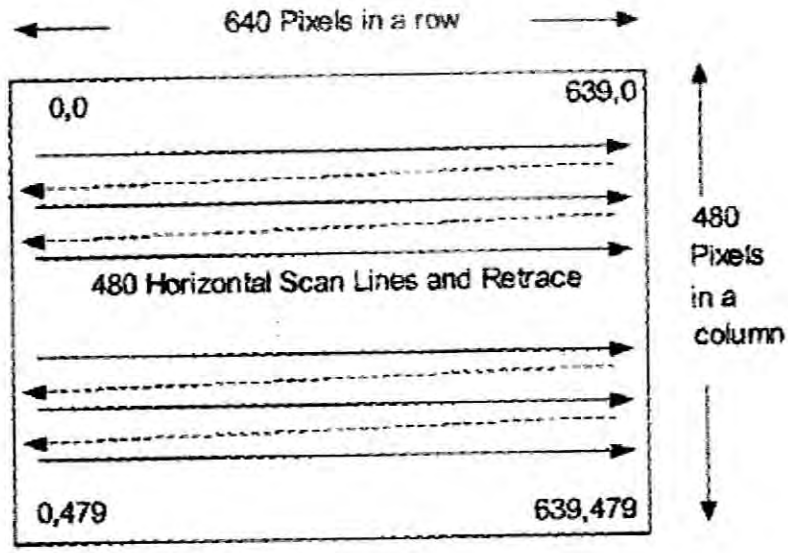

Figure 2.3: VGA Image – 640 x 480 pixels layout [3]

The screen refresh process seen in Figure 2.3 begins in the top left comer and paints l pixel at a time from left to right. At the end of the first row, the row increments and the column address is reset to the first column. Each row is painted until all pixels have been displayed. Once the entire screen has been painted, the refresh process begins again. The video signal paints or refreshes the image using the following process.

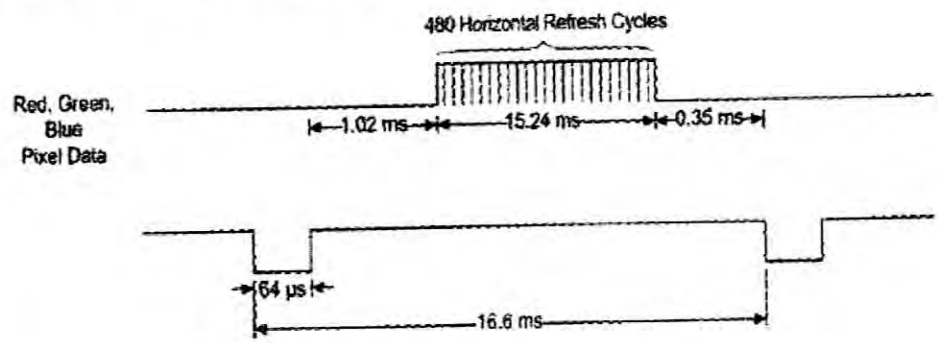

Figure 2.4: Vertical sync signal timing [3]

© Universiti Teknikal Malaysia Melaka

The vertical sync signal, as shown in Figure 2.4 tells the monitor to start displaying a new image or frame, and the monitor starts in the upper left comer with pixel 0, 0. The horizontal sync signal, as shown in Figure 2.5, tells the monitor to refresh another row of 640 pixels. After 480 rows of pixels are refreshed with 480 horizontal sync signals, a vertical sync signal resets the monitor to the upper left comer and the process continues. During the time when pixel data is not being displayed and the beam is returning to the left column to start another horizontal scan, the RGB signals should all be set to black color or all zero.

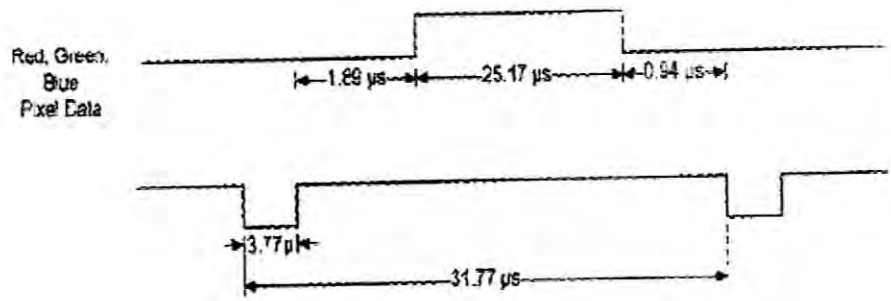

Figure 2.5: Horizontal sync signal timing [3]

Many VGA monitors will shut down if the two sync signals are not the correct values. Most PC monitors have an LED that is green when it detects valid sync signals and yellow when it does not lock in with the sync signals. In a PC graphics card, a dedicated video memory location is used to store the color value of every pixel in the display. This memory location is used to store the color across the screen to produce the RGB signals. There is not enough memory inside current generation FPLD chips for this approach so other techniques will be developed which require less memory.

To provide interesting output options in complex designs, video output can be developed using hardware inside the CPLD or FPGA. Only five signals or pins are required, two sync signals and three RGB color signals. A simple

© Universiti Teknikal Malaysia Melaka

resistor and diode circuit is used to convert TTL output pins from the CPLD to the analog RGB signals for the video signal. This supports two levels for each signal in the RGB data and thus produces a total of eight colors.

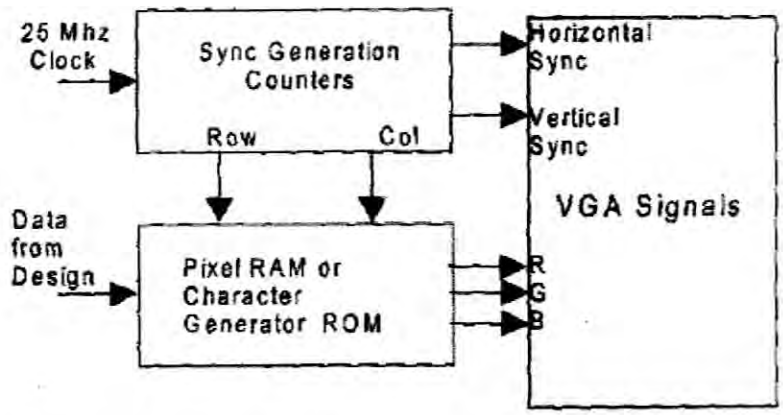

Figure 2.6: CPLD based generation of VGA video signals [3]

As seen in Figure 2.6, a 25.175 MHz clock, which is the 640 by 480 VGA pixel data rates of approximately 40ns is used to drive counters that generate the horizontal and vertical sync signals. Additional counters generate row and column addresses. In some designs, pixel resolution will be reduced from 640 by 480 to a lower resolution by using a clock divide operation on the row and column counters. The row and column addresses feed into a pixel RAM for graphics data or a character generator ROM when used to display text. The required RAM or ROM is also implemented inside the CPLD chip.

© Universiti Teknikal Malaysia Melaka

2.5 VHSIC HARDWARE DESCRIPTION LANGUAGE (VHDL)

In [4], VHDL or VHSIC Hardware Description Language, is commonly used as a design-entry language for field-programmable gate arrays and application-specific integrated circuits in electronic design automation of digital circuits. VHDL was originally developed at the behest of the US Department of Defense in order to document the behavior of the ASICs that supplier companies were including in equipment. That is to say, VHDL was developed as an alternative to huge, complex manuals which were subject to implementation-specific details.

The idea of being able to simulate these "documents" was so obviously attractive that logic simulators were developed that could read the VHDL files. The next step was the development of logic synthesis tools that read the VHDL, and output a definition of the physical implementation of the circuit. Modern synthesis tools can extract RAM, counter, and arithmetic blocks out of the code, and implement them according to what the user specifies. Thus, the same VHDL code could be synthesized differently for lowest cost, highest power efficiency, highest speed, or other requirements.

VHDL has a syntax that is essentially a subset of the Ada programming language, along with an added set of constructs to handle the parallelism inherent in hardware designs. VHDL is strongly-typed and case insensitive. The initial version of VHDL, designed to IEEE standard 1076-1987, included a wide range of data types, including numerical (integer and real), logical (bit and boolean), character and time, plus arrays of `bit` called `bit_vector` and of `character` called string.

A problem not solved by this edition, however, was "multi-valued logic", where a signal's drive strength (none, weak or strong) and unknown values are also considered. This required IEEE standard 1164, which defined the 9-value logic types: scalar `std_logic` and its vector version `std_logic_vector`.

© Universiti Teknikal Malaysia Melaka