FPGA Based Graphics Controller

Arun Babu
Asst. Professor, Dept. of ECE
Heera College of Engineering and Technology
Trivandrum, Kerala

***Abstract*—A graphics controller is a circuit which generates the output images to be fed into a display. Graphics controller is also known a Graphics Coprocessor. It is similar to a microprocessor, except that it is dedicated for video purpose. The graphics controller processes the graphics received by the computer and creates dots and lines on the screen to generate the picture. In this paper, the architecture and implementation of a reconfigurable graphics controller using Xilinx FPGAs is presented. This graphics controller is capable of displaying multiple resolutions namely VGA, SVGA, XGA and WXGA respectively. The architecture of this controller is designed such that it can be modified very easily and according to the requirement. The coding of the system is done in Verilog. The design is selected after synthesizing with different synthesizing engines and taking the most optimized version. The design can also be used in FPGAs by other vendors. The Graphics controller was tested in Xilinx Spartan 3E Evaluation board. The end usage of the controller is in embedded systems and FPGA based computers.**

Keywords*: FPGA, Graphics Controller, Xilinx, Verilog, VGA*

I. INTRODUCTION

VGA (Video Graphics Array) is one of the most used display interface in many systems such as Personal Computers, LCD projectors, Media players, CCTV systems etc. VGA [1] standard is widely employed in embedded systems. This is one of the earliest as well as easiest of the display standards. VGA is an analog display standard owing to the analog color signals. VGA is a standard used for showing images or information on a VGA display device. VGA Controller [2] is a system to generate the essential timing signals and to interface the digital video data with the display device. The design is done so as to achieve high flexibility. The controller is capable of displaying four resolutions VGA, SVGA, XGA, and WXGA. The system can be programmed to work in any color bit mode less than or equal to 24bits. Video DAC [3][4] for VGA out is also designed. A built in self-test (BIST) [5] functionality has been added into the device for improved flexibility. The design has been synthesized and implemented on a Xilinx Virtex 5 (XUPV5LX110T) FPGA device using Xilinx ISE 13.2 Design Suite. The HDL used is Verilog [6]. The design can be used with any choice of FPGA.

The resolution and test mode are selectable too. A comparison study of codes synthesized using different synthesize engines are discussed.

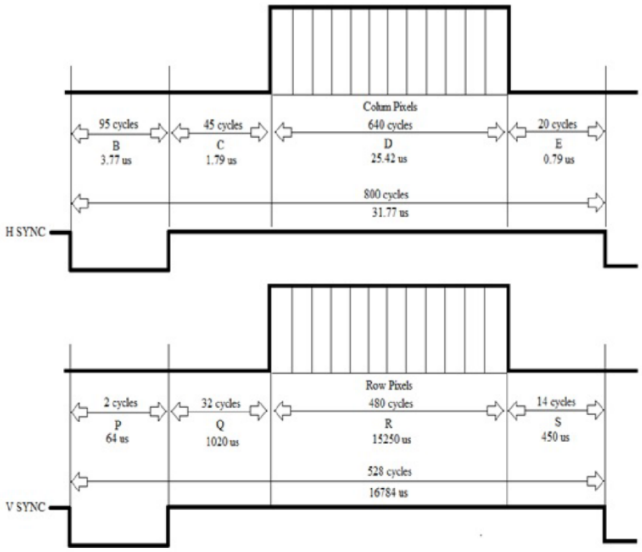


FIGURE I: VGA TIMING DIAGRAM

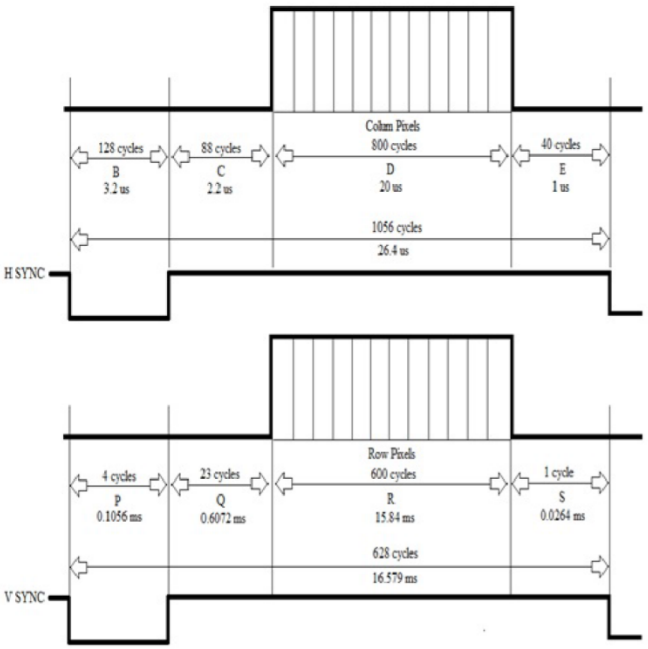


FIGURE II: SVGA TIMING DIAGRAM

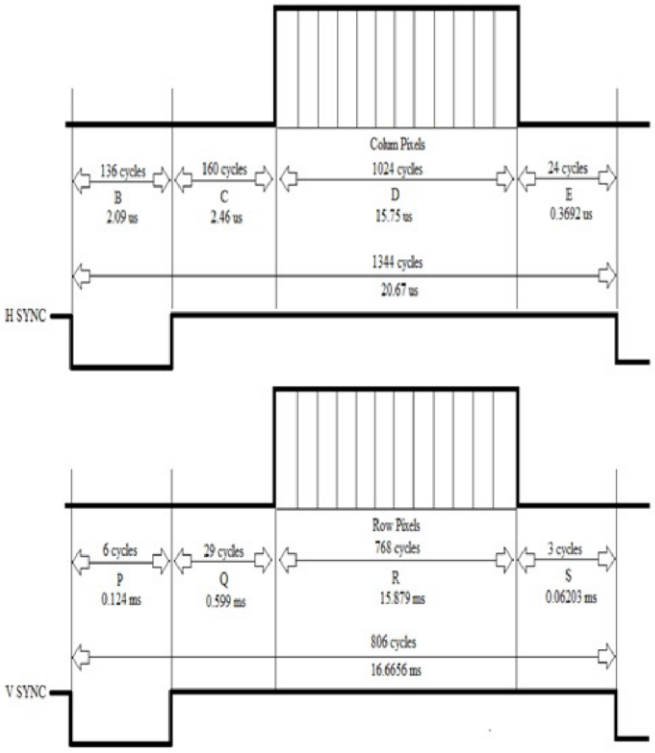
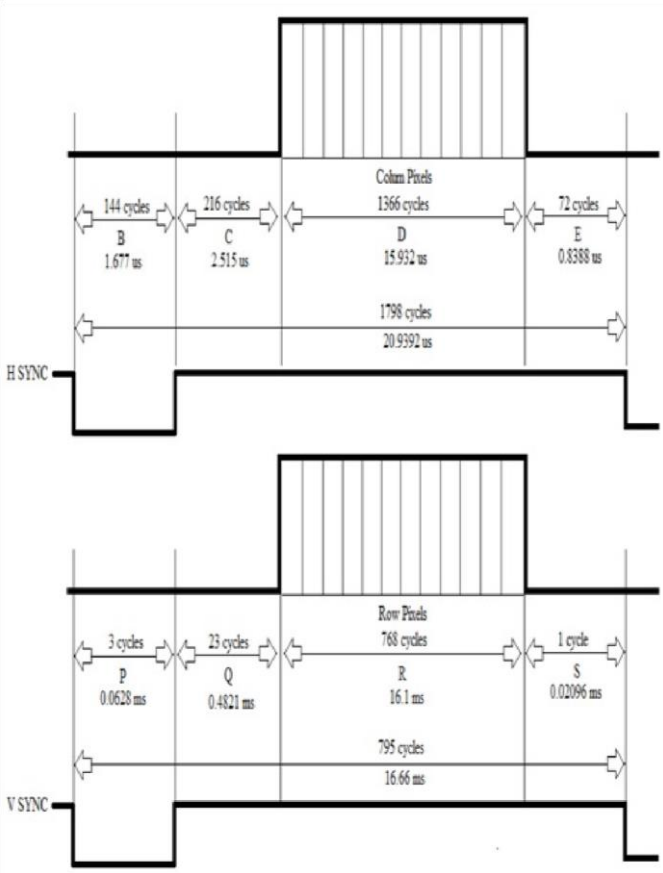


FIGURE III : XGA TIMING DIAGRAM



Sl No.	Resolution @ 60Hz	Clock frequency (MHz)
1	640 x 480(VGA)	25.175
2	800 x 600(SVGA)	40
3	1024 x 768(XGA)	65
4	1366 x 768(WXGA)	85.86

FIGURE V CLOCK FREQUENCIES

II ARCHITECTURE AND EXPLANATION

The architecture is chosen such that it can be used in small or large embedded systems as well as in FPGA based computers [10]. A custom architecture is used here.

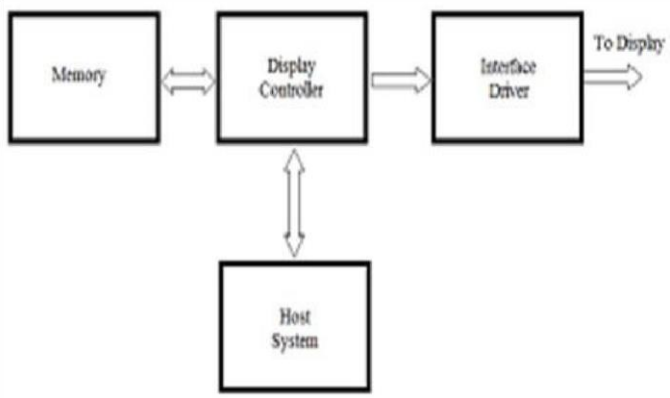


FIGURE V SYSTEM ARCHITECTURE

The graphics controller is built round four basic blocks. They are Host System, memory, Display Controller and Interface Driver. No bus is used, for simplicity.

HOST SYSTEM

Host system is the one, which provides the system with the visual data. It can be an embedded microcontroller, microprocessor or an FPGA itself. Host system is interfaced directly with the graphics controller. Host system can give color data bits less than or equal to 24bits width.

DISPLAY CONTROLLER

Display Controller is the block which interface s the color bits with the memory, generates the timing signals and interfaces the color data with the interface driver. This block accepts the color information from host system stores the data on to the memory and takes the data and output to the interface driver.

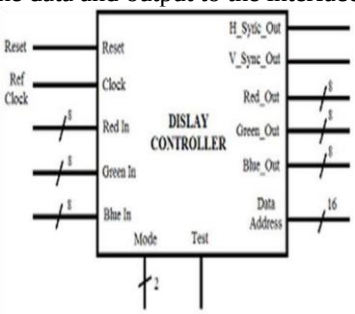


FIGURE IV : WXGA TIMING DIAGRAM

Reset signal resets the whole system. Ref clock is the master clock. Mode bits are used to select the resolution. Test bit is used to select whether the system should work in test mode or normal mode. Hsync out and Vsync out are the horizontal and vertical sync signals respectively. Xilinx DCM has been used for generating the required clock frequencies for each resolution.

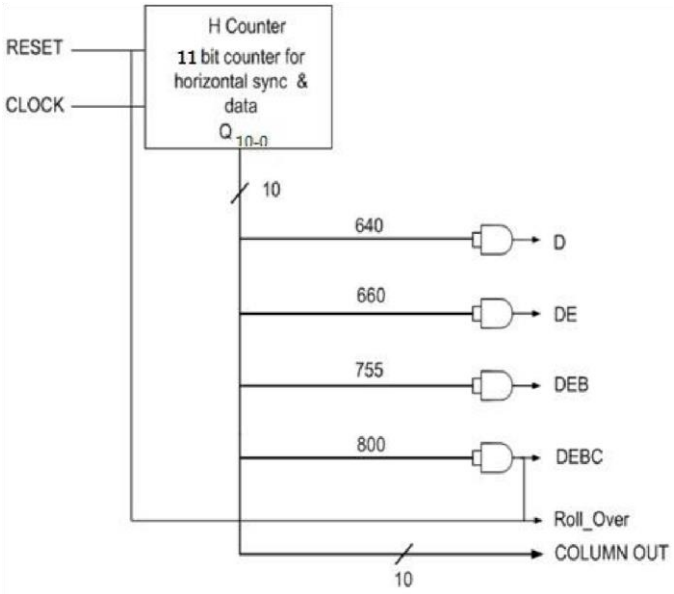


FIGURE VII HORIZONTAL COUNTER

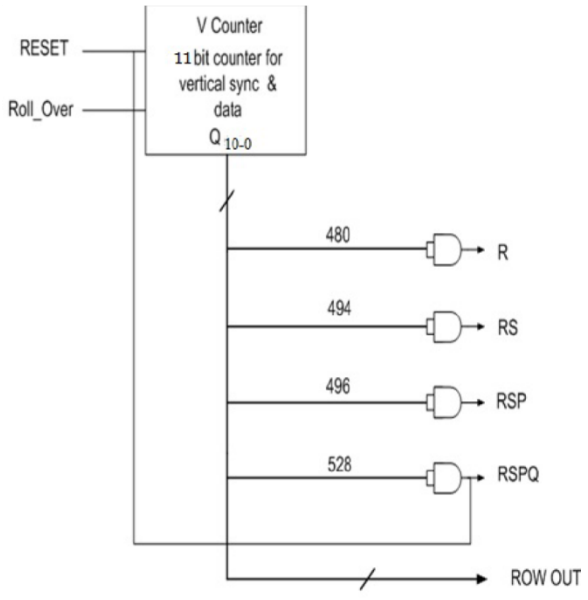


FIGURE VIII VERTICAL COUNTER

When inactive, both the synchronization signals are at a logic one value. The horizontal sync signal when it goes low for 3.77 us, then it masks the beginning of the row scan. A 1.79 us high follows this. Next the data for the 3 colour signals are sent, one pixel at a time for 640 columns for 25.42 us. Finally after the last column pixel, there is another 0.79 us of inactivity on RGB signal lines for the

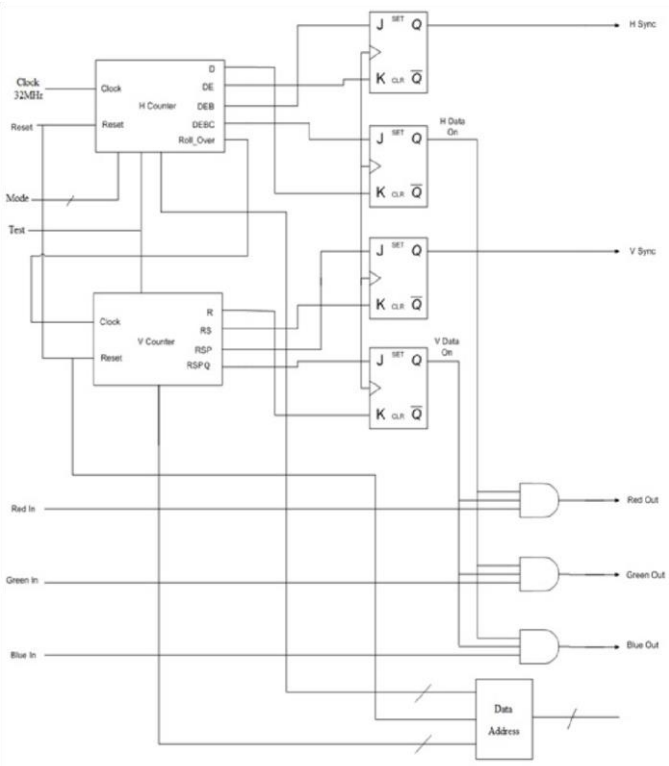


FIGURE IX DISPLAY CONTROLLER INTERNAL DIAGRAM

horizontal retrace before the horizontal sync goes low again for the next row scan. The total time to complete one row scan is 31.77us. The timing for the vertical sync signal is analogus to the horizontal one. The 64 us active low vertical sync signal resets the scan to the top left corner of the screen. A 1020 us high follows this on the signal. Next, there are the 480, 31.77us row scans, giving a total of 15250 us(480x31.77), as shown in section R in the timing diagram og Vertical portion of VGA.

Finally after the last row scan, there are another 450 us before the vertical sync goes low again to start another complete scan in top left corner. So a total of 16,784 us is taken to complete a total scan. To get the monitor operating correctly, we simply have to get the horizontal and the vertical sync signals timing correct and then send out RGB data for each pixel at the right column and row position. Eg: If you want to turn on the red pixel at row 18, column 56, simply wait for the scan to reach row 18 and column 56 and then set the red pixel to the logic one value. To accomplish this , we have to generate the horizontal and vertical timing signals correctly. Also have to keep track of the current row and column to know where the current scan is. For this purpose 2 counters are used. One for generating the horizontal sync and keeping track of the column count and the other is for generating the vertical sync and keeping track of the row count.

For a VGA resolution of 480 x 640 screen resolution, we use a clock with a 25.175 Mhz frequency. We get this

frequency by multiplying the total cycles for row and the column into the screen refresh rate used.

Ie: 800 x 525 x 60Hz = 25.2 Mhz app. Therefore, period for 25.175 Mhz clock = 1 divided by (25.175 multiplied 10 power 6) = 0.0397 us per clockcycle. Therefore for section B = 3.77divided by .0397 = 95 clock cycles. Similarly we get the number of cycles for each section. This is for VGA. If we want to do for the other resolutions, we have to just substitute the values of this VGA resolutions with the corresponding values of each resolutions. The counter keeps on counting and when the count reaches the corresponding end values of each sections, a high value is given as output usin AND gates and the output values are send as either inputs of J-K ipops whose outputs are the Horizontal sync, Vertical Sync, Horizontal Data on and the Vertical Data on. And according to these outputs the pixel datas are read and is fed as output signals. In this design, the memory used is the internal Block RAM of Xilinx FPGA. It is a static RAM. An external static ram or SD RAM can also be used. External memory use will require additional memory interfacing circuits. Block and distributed

RAM's on Xilinx FPGA's The configuration logic blocks(CLB) in most of the Xilinx FPGA's contain small single port or double port RAM. This RAM is normally distributed throughout the FPGA than as a single block (It is spread out over many LUT's) and so it is called "distributed RAM". A look up table on a Xilinx FPGA can be configured as a 16*1bit RAM , ROM, LUT or 16bit shift register. This multiple functionality is not possible with Altera FPGA's. For Spartan-3 series, each CLB contains up to 64 bits of single port RAM or 32 bits of dual port RAM. As indicated from the size, a single CLB may not be enough to implement a large memory. Also the most of this small RAM's have their input and output as 1 bit wide. For implementing larger and wider memory functions you can connect several distributed RAM's in parallel. Fortunately you need not know how these things are done, because the Xilinx synthesizer tool will infer what you want from your VHDL/ Verilog code and automatically does all this for you. A block RAM is a dedicated (cannot be used to implement other functions like digital logic) two port memory containing several kilobits of RAM. Depending on how advance your FPGA is there may be several of them. For example Spartan 3has total RAM, ranging from 72 kbits to 1872 kbits in size. While Spartan 6 devices have block RAM's of up to 4824 Kbits in size. Interface driver will interface the color data signals with the display. Since VGA is an analog standard, the digital color signals have to be converted to analog form with peak value of 0.7V. A resistive binary R-2R DAC is used as interface driver) Video DAC). Dedicated video DAC chips like ADV7123 can also be used instead of the video DAC which ensures perfect reproduction of the image. VGA is an analog display standard. The output from FPGA is digital voltage of 3.3V. So, we have to convert the digital signals to analog signals of maximum amplitude of 0.7V. R 2R ladder or ADV DAC chip can be used as the digital to analog converter [9]. The ADV7123 (ADV) is a triple high speed, digital-to-analog converter on a single monolithic chip. It consists of three high speeds, 10-bit, video

D/A converters with complementary outputs, a standard TTL input interface and a high impedance, analog output current source. The ADV7123 has three separate 10-bit-wide input ports. A single +5 V/+3.3 V power supply and clock are all that are required to make the part functional. The ADV7123 has additional video control signals, composite SYNC and BLANK .The ADV7123 also has a power-save mode. The ADV7123 is fabricated in a +5 V CMOS process. Its monolithic CMOS construction ensures greater functionality with lower power dissipation. The ADV7123 is available in a 48-lead LQFP package.

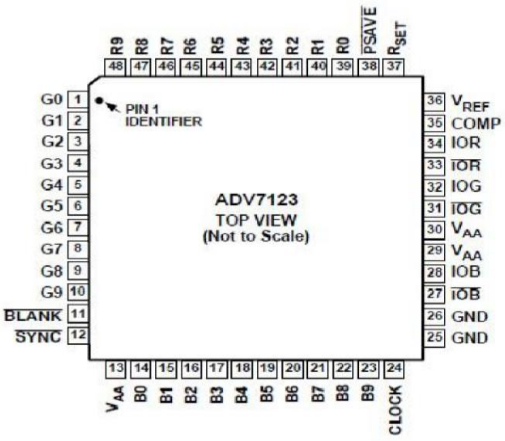


FIGURE X ADV 7123 PIN DIAGRAM

The ADV7123 contains three matched 10-bit D/A converters. The DACs are designed using an advanced, high speed, segmented architecture. The bit currents corresponding to each digital input are routed to either the analog output (bit = 1) or GND (bit = 0) by a sophisticated decoding scheme. As all this circuitry is on one monolithic device, matching between the three DACs is optimized. As well as matching, the use of identical current sources in a monolithic design guarantees Monotonicity and low glitch. The onboard operational amplifier stabilizes the full-scale output current against temperature and power supply variations.

FIGURE XI XILINX DCM INTERFACE

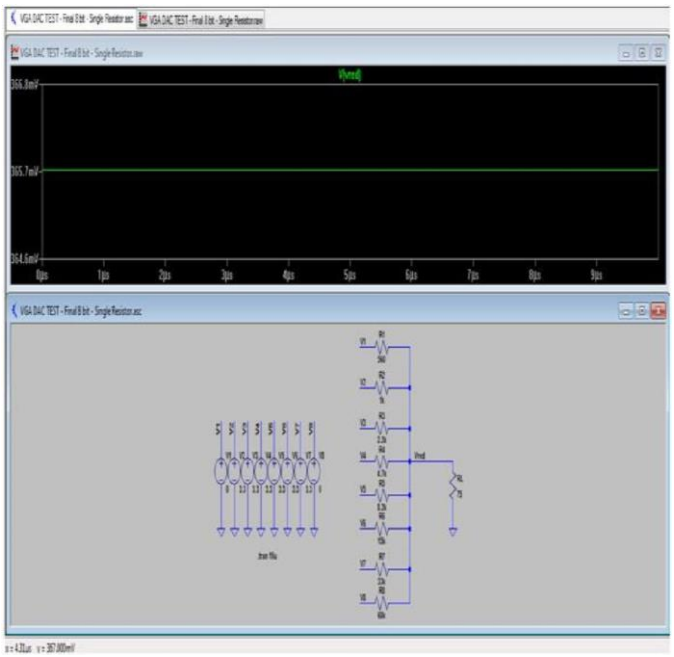


FIGURE XII DAC DESIGN VERIFICATION USING LTSPICE

IV EXPERIMENTAL RESULTS

The Graphics Controller has been coded in Verilog, which is the IEEE 1364 standard. Xilinx ISE 14.1 is used for writing

the RTL code, Xilinx XST is used for synthesizing and Modelsim 6.5 simulator is used for functional simulation. The hardware is implemented in Xilinx Virtex 5 (XUPV5LX110T) evaluation board manufactured by Digilent Inc. The design has been synthesized with different constraints with timing constraints given high priority. Optimization of timing is the critical constraint in this design.

FIGURE XIV TEST OUTPUT

Device Utilization Summary			
Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	109	69,120	1%
Number used as Flip Flops	109		
Number of Slice LUTs	241	69,120	1%
Number used as logic	240	69,120	1%
Number using O6 output only	214		
Number using O5 output only	18		
Number using O5 and O6	8		
Number used as exclusive route-thru	1		
Number of bonded IOBs	83	640	12%
Number of BlockRAM/FIFO	32	148	21%
Number using BlockRAM only	32		
Number of 36k BlockRAM used	32		
Total Memory used (KB)	1,152	5,328	21%

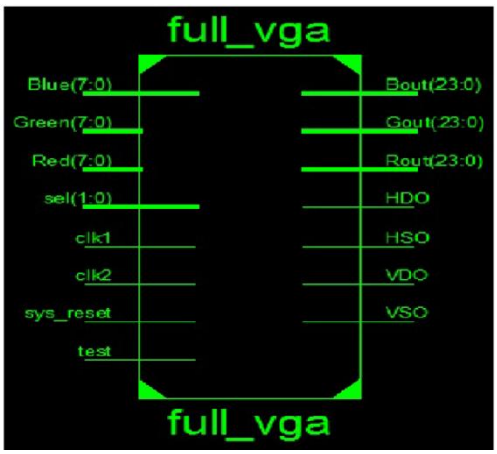
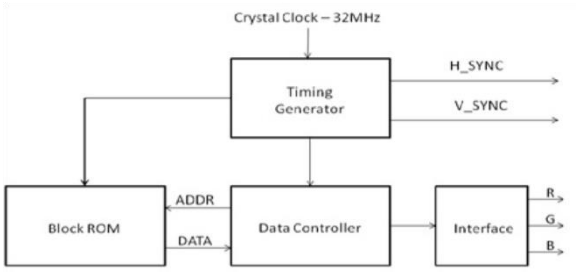


FIGURE XII RTL SCHEMATIC
FIGURE XV DEVICE UTILIZATION



A graphics Controller based on Xilinx FPGA is designed and implemented and the result is verified. Using efficient optimization techniques, the timing constraint is successfully met. Built-in-self-test is successfully integrated into the system. Four different resolutions are selectable using Mode bits. A Video DAC using resistors have been designed. A flexible architecture has been used for the implementation. The design can be easily adapted for use with other FPGAs. Efficient use of Xilinx DCM is done to generate different frequencies from a master frequency. The graphics controller can further be extended for use with other display methodologies like DVI, HDMI etc.

V REFERENCES

[1] Thompson S, VGA - sign choices for a new video subsystem, IBM Systems Journal, 1988, 185-197 subsystem, IBM Systems Journal, 1988, 185-197

[2] Van-Huan Tran, Xuan-Tu Tran, An E_cient Architecture Design for VGA Monitor Controller, IEEE, 2011,3917-3921.

[3] Zoran Mijanovic, Rada Dragovic-lvanovic, Wubiga Stankovik, R2R Digital-Analog Converter (DAC), IEEE Instrumentation and Measurement Technology Conference, Brussels, Belgium, June 4-6, 1996, 1034-1039

[4] Michael Peter Kennedy, On the Robustness of R-2R Ladder DACs, IEEE Transactions On Circuits And SystemsPart I: Fundamental Theory And Applica- tions, Vol. 47, No. 2, February 2000, 109- 116. [5] Sunil R Das, Getting Errors to Catch ThemselvesSelf- Testing of VLSI Circuits With Built-In Hardware, IEEE Transactions On Instrumentation And Measurement, VOL. 54, NO. 3, JUNE 2005, 941-955

[6] Carl Ebeling, Brian French, Abstract Verilog: A Hardware Description Language for Novice Students, IEEE International Conference on Microelectronic Systems Education, 2007.

[7] A.Moreno, E.John, A Flexible Design For Timing Signals Generation For The Conversion Of Computer Video Formats To Sdtv 480P, IEEE, 1999.

[8] Zaheeruddin Syed, Munwar Shaik, FPGA Implementation OF VGA Controller, International Conference on Electronics and Communication Engineering (ICECE), Pune, 16th Sept, 2012

[9] Marche. D, Modeling R-2R Segmented-Ladder DACs, IEEE transactions on Circuits and Systems I, Jan 2010, 31-43.

[10] Felix Huber, Peter Behr, Hans-Peter Roser, Samuel Pletner, ,FPGA based On-Board Computer System for the Flying Laptop Micro-Satellite, Steinbeis Transferzentrum Raumfahrt, Universitt Stuttgart, Fraunhofer Institute for Computer Architecture and Software Technology