VERIFICATION OF VGA PROTOCOL BY USING SYSTEM VERILOG

C SREESANDHYA[1] **K RAMESH[2]**
[1]PG Scholar, VLSI,SREE VIDYANIKETHAN ENGINEERING COLLEGE , SREE SAINATH NAGAR, CHITTOOR, ANDHRA PRADESH.
[2]Assistant Professor, Department of ECE, SREE VIDYANIKETHAN ENGINEERING COLLEGE , SREE SAINATH NAGAR, CHITTOOR, ANDHRA PRADESH.

ABSTRACT: The goal is to design VGA protocol using Verilog and verify it by using system Verilog code. VGA (video graphic array) is used as a standard display interface. This protocol is used to ensure the portability with any user.in order to design and verify the VGA protocol Verilog code is used based on IEEE standards.the standard monitor screen resolution of 800*525 but display screen resolution is 640*480 the total number of pixels present in the screen is 307,200 (640*480=307200). Goal is to find the Address for the every pixel in the image.

Watch words: Modelsim 6.6d, Questa Sim 10.0b.

INTRODUCTION

VGA (video graphic array) protocol is standard display interface. Which contains 640columns and 480rows of picture elements called pixel it displays the video, image or information by simply connecting the system with a monitor.VGA connector is having three rows and 15-pins. Pin (P) contact is called as female DE-15 connector shown in fig1.1 and socket(S) contact is called as male DE-15 connector shown in fig1.2. Fig1.1 shows D is the prefix for the whole series because the pin contact is D shaped series. A,B,C,D or E denote the shell size then it is denoted as a DE-15 connector.P and S denote the gender of the part.

Figure 1.1 female DE-15 connector

Figure 1.2 VGA cable with DE-15 male connector

System Verilog is combined hardware description language and hardware verification language system verilog is extension to verilog HDL. System verilog is able to design and verification it is easy to design and 95-100% bugs and combinations are covered in system Verilog. To verify the address of the every pixel in the image the RGB data must be take from the image file in bitmap format using MATLAB and rearrange the data using Microsoft excel and store in MIF file. after the VGA protocol program was written in verilog HDL compile using Questa sim software after compilation completed simulate the program and check the proper waveforms are coming are not and check the same address is coming for pixels by comparing with MIF file stored in Microsoft Excel. And create the system verilog verification environment for the VGA protocol compile and simulate it by using Questa sim software after the simulation is succeed and verify the verification environment output results with design code results i.e, matched or not matched.

VGA PROTOCOL

The standard monitor screen resolution for VGA is 640columns and 480rows of picture elements called pixels.(640*480=307200) 307200 pixels are present in this screen by turning on and off the each pixel separately we can generate the image. By gathering the more pixels an image is displayed on the screen not by turning on one pixel. Video monitor uses CRT(cathode ray tube) when a clolour TV receives the video signal it decodes the signal into 3 separate signals one for each other primary colour RGB. Inside the TV tube they are three electron guns are available each electron gun emits electron beam it scans the entire screen row by row on the face of the screen phosphors dots are available. Shadow mask is constructed between the electron gun and screen holes are arranged on the shadow mask by this holes the red electron beam red electron gun

can only bombard with red phosphors dots on the screen then green and blue electron beams also can bombard with their respective phosphors dot on the face of the screen by this we can generate the continuous image on the screen

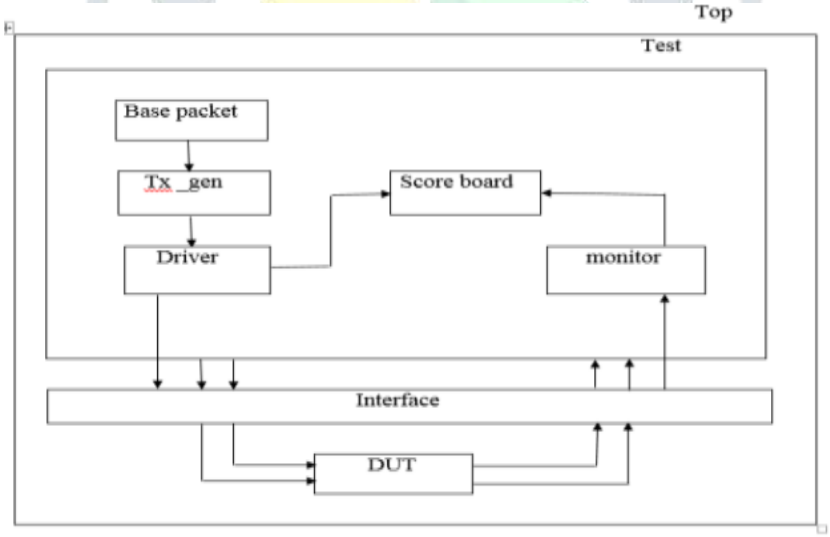
Fig2.1 shows scanning process the scanning process starts from top left corner at row0, column0 of the screen. The electron beam scan from row0, column0 to the right when it reaches to the end of the column it again retrace to the next row1 and starts the scanning it is called horizontal scanning. When electron beam reaches to the bottom right corner it again retrace to the first row0, column0 it is called vertical scanning and again scanning process is repeated. Red, Green, Blue (RGB), horizontal and vertical sync signals these 5 active signals control the VGA protocol. Red green and blue these three colours are collectively referred as RGB signal,control the colour of the pixel at the given location on the screen.RGB signals are analog signals from 0.7 to 1.0 at this analog level only we can control this colours.by changing this analog level all other colours are produced. To determine the time to scan a row horizontal sync signal is used, and vertical sync signal is used to determine the time to scan entire screen. By manipulating these 5 signals an image is displayed on the screen. In order to get the 640*480 screen resolution 25.175MHz clock frequency Is used. Clock frequency is increased for higher screen resolution. 1/25.175MHz = 0.0397us per clock cycle.

Fig3.1shows The standard monitor total screen resolution is 800*525 but display screen resolution is 640*480 the electron beam scan down the entire screen row by row.the electron beam for horizontal scan it goes linearly during the display region it agin goes low it is retrace time i.e when the electron beam scan linearly when it reaches to the right end of the column it again retrace to the next row.the horizontal sync pulse goes high at the display region is called back porch,at the retrace time horizontal sync pulse goes low is called front porch.the same saw tooth waveform for vertical sync pulse also but timings are different.

Referred fig3.2 for the horizontal synchronization at section1 3.77us signal is used which is approximately 95 clock cycles(3.77/0.0397),at section2 1.79us signal is used which is approximately 45 clock cycles.at section3 25.17us window which is approximately 640 pixels at this period only we can generate the image or video .and at section4 0.79us Is used which is approximately 20 clock cycles.at 25.17us signal only we can read the image or video other signals are blank is called blanking period.total number of clock cycles needed for horizontal synchronization is 800(95+45+640+20).for vertical synchronization at 15.25us signal only we can generate image or video 480 clock cycles are needed.at section p 64us signal is used which is approximately 2clock cycles.at section q 32clock cycles are needed, similarly at section s 14clock cycles are needed.at 15.25us signal only we can read the image and other signals are blank is called blanking period in vertical synchronization.total number of clock cycles needed is 525(2+32+480+14).

Block diagram fig 4.1 shows inputs are clock and reset signals, output signals are RGB (red, green, blue),h_count,v_count,h_sync,v_sync and blanking signals. To get the 640*480 resolution 25MHz clock frequency is used. The clock generator reduce the clock frequency from 50MHz to 25MHz, vga_sync block is used to generate the timing and synchronization signals.h_count and v_count signals are used to find the current location of the pixel,h_sync signal is used to determine the required time to scan a row.v_sync signal is used to determine the required time to scan the entire screen. To find the address for the image data block address generator block is used it find the address by using h_sync and v_sync signals. Image data block will get the index data from MIF file according to the address generated block. Image data block is connected to the image index block it is used as the address. The image index block will get the RGB data each one consist of 8_bit data where as R_data[23:16], G_data[15_8] and B_data[7:0] respectively.

VERIFICATION ENVIRONMENT



Verification environment is group of class's performing specific operation. Fig shows through base packet we are giving give the only input values by tx_gen we are generating stimulus. The driver receives the data from tx_gen and send these signals using interface to DUT. The monitor and it displays the various messages according to the operation. Monitor sends the actual output to scoreboard and driver send the expected output. Scoreboard verify the actual and expected results are matched or not.

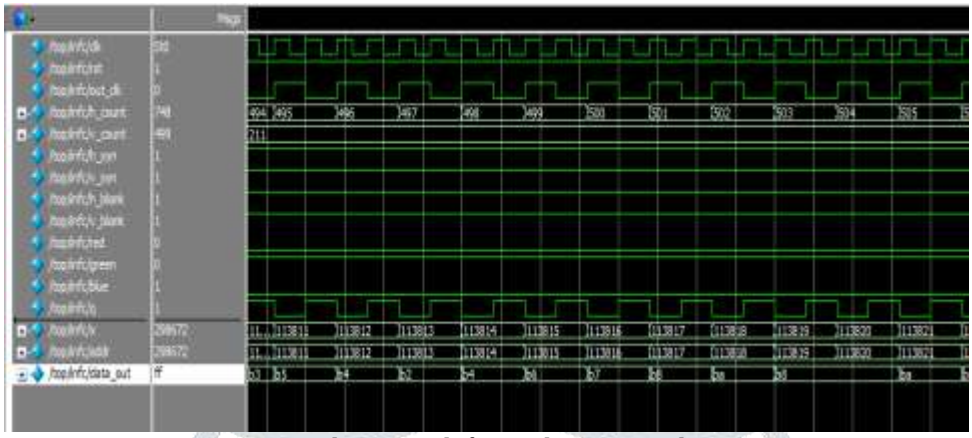
Test is the test layer of the verification specification and work as the functional block is to be created. Under the base packet, tx_gen, driver, monitor and scoreboard are using the create the design of the modules to generate the signal and output are to check the monitor and scoreboard.

VERIFICATION

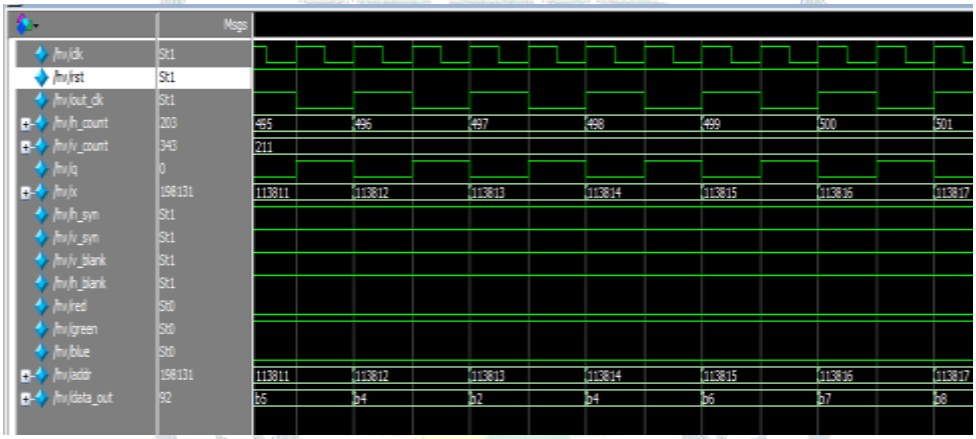
The VGA protocol program was written in verilog HDL compile using Questa sim10.0b software after compilation completed simulate the program and check the proper waveforms are coming are not and check the same address is coming for pixels by comparing with MIF file. And

create the system verilog verification environment for the VGA protocol compile and simulate it by using Questa sim software after the simulation is succeed and compare the output results with verilog HDL results.

SIMULATION RESULTS:



Simulation result for verilog HDL code



Simulation result for system verilog verification environment

CONCLUSION

System verilog language is extension to verilog HDL it is easy to design 95_100% bugs and combinations are covered. System verilog verification environment is created by this address is found for every pixel and address is matched for every pixel in the image by comparing with verilog HDL proper output waveforms are obtained and verified.

REFERENCES:

[1]Altera corporation.”introduction to Quartus II” (28 February 2012),[online].
Available:http://users.ece.gatech.edu/~hamblen/UP3/intro_to_quartus2.pdf
[2]Altera corporation, “what is an FPGA?” (26 october 2011), [online]. Available:http://www.altera.com/products/fpga.html
[3]E.Hwang,”Build a VGA Monitor Controller",Nov.2004,[Online].Available:www.circuitceller.com.
[4]W.Wolf,”FPGA-based system design”,Pearson education, Inc,USA,2004.

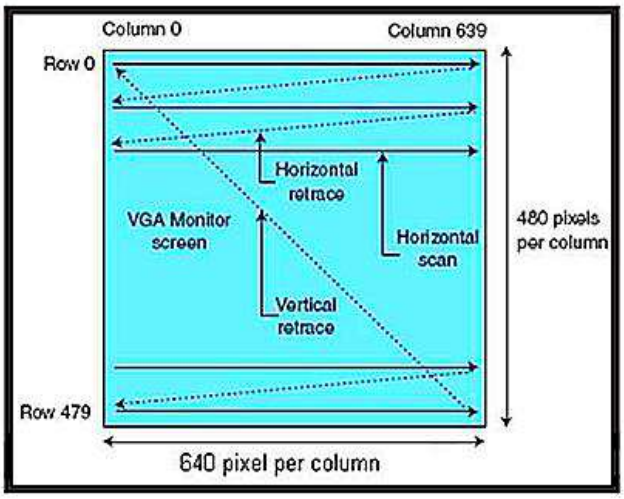


Figure 2.1 scanning pattern of VGA protocol

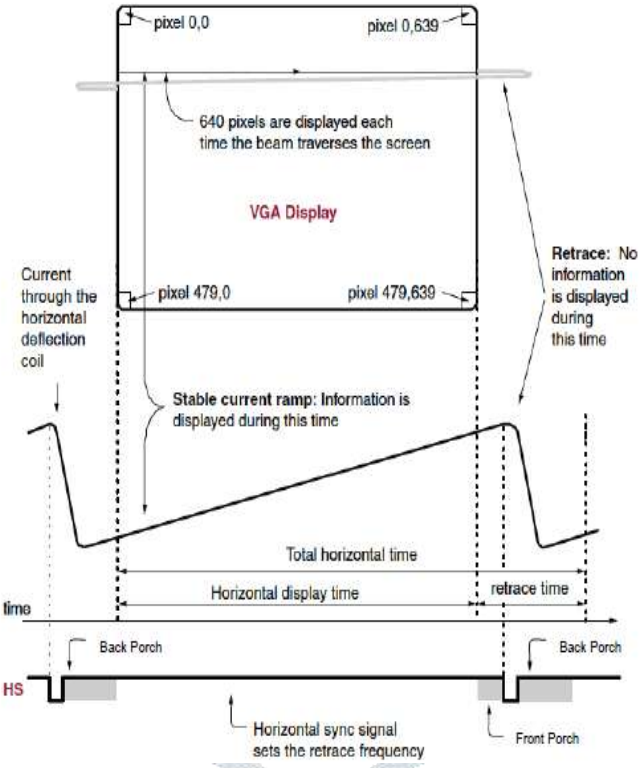


Figure 3.1 Timing diagram of VGA protocol

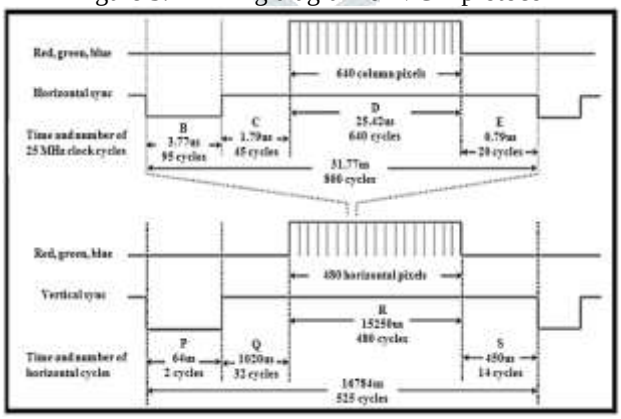


Figure 3.2 Timing diagram

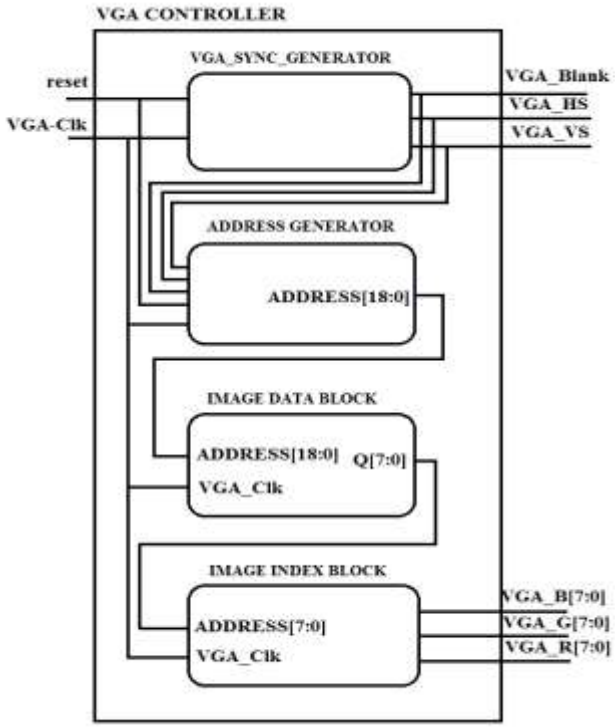


Figure 4.1 VGA protocol block diagram