VOL. 11, NO. 15, AUGUST 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

REAL TIME FPGA IMPLEMENTATION FOR VIDEO CREATION VGA FRAME RATE CONVERSION SYSTEM

Umarani E. and Vino T.
Sathyabama University, Chennai, India
E-Mail: umaeswaran132@gmail.com

ABSTRACT

This paper introduces real time Full VGA display Frame rate conversion implemented on FPGA. By using the Frame rate conversion, we can display the Video game. This article gives a programming design of video game based on the FPGA using VHDL. The Game realized the function of the movement and rotation of blocks, randomly generating next blocks. The successful transplant of video game provides a template for the development of other visual control systems in the FPGA. This system improves the Quality of Video, can create Images and providing Animation to the Images and can control the system using Joy Stick, code in VHDL Language. Planning to add Encryption with the created video for Communication.

Keywords: VGA, video creation, VHDL, Sync pulses, image.

INTRODUCTION TO VIDEO PROCESSING

DEFINITION OF VIDEO SIGNAL

Video signal can be fundamentally any sequence of time varying images. A still image is a spatial distribution of intensities that persist constant with time, whereas a time varying image has a spatial intensity distribution that varies with time. Video signal is considered as a series of images called frames. An illusion of continuous video is by changing the frames in a quicker manner which is mostly termed as frame rate. Analogue Video Signals Despite the advance of digital video technology, the most usual consumer display structure for video still uses analogue display devices may be CRT. Until all earthly and satellite broadcasts become digital, analogue video formats will remain significant. The three main Analogue Video Signal formats are: NTSC (National Television Systems Committee), PAL (Phase Alternate Line) and SECAM (Sequential Color with Memory). All the 3 are television video formats in which the information in each picture is grabbed by CCD or CRT is scanned from left to right to create a sequential intensity signal. The formats take opportunity of the determination of human vision by using interlaced scanning pattern in which the odd and even 13, two lines of each picture are read out in two separate scans of the odd and even fields respectively. This allows good reproduction of movement in the scene at the proportion to low field rate of 50 fields/sec for PAL and SECAM and 60 fields/sec for NTSC.

We are creating an image in which we are providing an animation. For this project the image of the game has been created which contains a wall, a ball and a bar.

In this project, video creation can be done based on the FPGA using HDL. The game is kind of Pong Game, the game contains a wall, a ball and a bar. The Game realized the function of the movement and rotation of blocks, made generating next blocks. The successful transplant of video game provides a base for the development of other visual control systems in the FPGA. By using this system we can create Images and provide animation to the Images and can control the system using Joy Stick, code in VHDL Language.

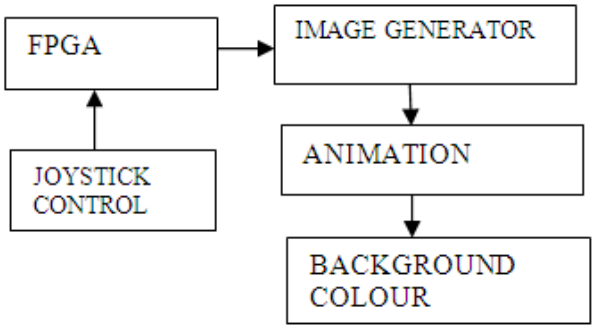

Figure-1. Flow diagram.

TECHNIQUES USED

DE Interlacing

De-interlacing is the activity of converting interlaced video, such as common analog television signals.

Interlaced video frames are of 2 fields taken in sequence, each subsequent scanned at odd and even lines of the image sensor; analog television employed this technique because it allows for low transmission bandwidth and matches the properties of CRT screens.

However, most current displays are permanent progressive; hence the two fields need to be combined to a single frame, which tends to various visual defects which the de-interlacing process should try to avoid.

Frame rate conversion

Frame rate is also known as Frame frequency. It is the frequency (rate) at which an imaging device displays consecutive images called Frames.

The terms applies equally to film and video cameras, computer graphics and motion capture systems.

Frame-rate conversion is necessary almost always. It is needed in cinema when projecting 24fps material at 48fps/ 72fps. In future it comes into play when combining material shot at different frame rates or changing speeds for effect. It is used to reduce flicker for high frame-rate displays. And, it is useful when sending to different states and display devices.

This six-frame sequence tells how various types of motion vector-based processing can affect image quality The convention is that motion imaging has a native frame rate - the number of images created over a given period of time. When a ramp up / ramp down is done during creation, the native frame rate must be specified as it is no longer constant throughout. Also, time-lapse/high-speed filming is done with a planned presentation frame rate different from the capture frame rate.

FRAME RATE AND HUMAN VISION

The secular sensitivity and resolution of human vision varies depending on the type and characteristics of visual stimulus, and it varies between individuals. The human visual system can process 10 to 12 separate images per second and perceive them separately, and sequences at higher rates are kept as motion. Modulated light (such as a computer display) is kept as stable by the majority of participants in studies when the rate is higher than 50 Hz through 90 Hz. This perception of regulated light as steady is known as the flicker fusion threshold. However, when the regulated light is non-uniform and holds an image, the flicker fusion threshold can be much higher. With regard to image recognition, people have been recognizing a specific image in an unbroken series of different images, each of which lasts as little as 13 milliseconds. Persistence of vision sometimes accounts for very short single-millisecond visual stimulus having a duration of between 100 ms and 400 ms. Multiple stimuli that are very short are sometimes perceived as a single stimulus, as a 10 ms green flash of light immediately followed by a 10 ms red flash of light perceived as a single yellow flash of light.

MODULES

There are two type of section,

1. SYNC Module
2. IMAGE Generation Module

First module called SYNC MODULE will generate the

i) Horizontal Synchronization
ii) Vertical Synchronization and
iii) Control Signals

Second module called IMG GEN will create all the necessary images is used in the pong game

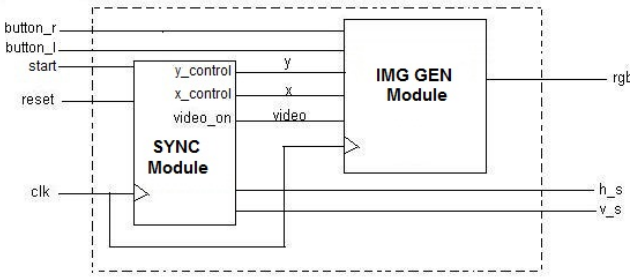

Figure-2. Module for sync and image generator.

SYNC MODULE

First we have designed the sync module. This module generates the horizontal sync and vertical sync for VGA video. In addition, sync module also gives control signals for the IMG GEN (image Generation) module.

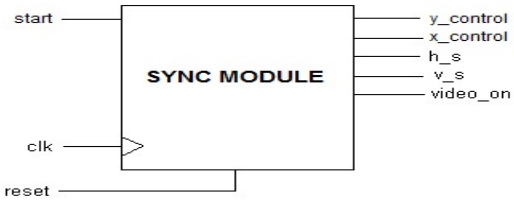

Figure-3. Sync Module.

SYNC module has 1 input and 2 output signals. The following shows the definition of these signals.

Input Signals clk:

System clock [25.175MHZ]

Output Signals:

H_S: Horizontal sync,
V_S: Vertical sync,

X_counter: output of a counter which places in the sync module. It counts from 0 to 639 (horizontal resolution of the VGA video =640)

Y_counter: output of a counter which places in the sync module. It counts from 0 to 479 (vertical resolution of the VGA video)s

Video_on: is used to enable/disable the video

VGA TIMING

To achieve 640x480 resolution we can set some constants in the program.

Abbreviation	Definition	Value
HR	Horizontal Resolution	640
HFP	Horizontal Front Porch	16
HBP	Horizontal Back Porch	48
HRet	Horizontal Retrace	96
VR	Vertical Resolution	480
VFP	Vertical Front Porch	10
VBP	Vertical Back Porch	33
VRet	Vertical Retrace	2

Figure-4. VGA terms.

Totally size of the monitor is 800x550 resolution. But display area is 640x480 resolution.

In our project we can use 50 Mhz as a system clock. But as per the list given previous part we need 25 Mhz clock signal in order to produce 640x480,@60Hz video. Hence we can use 2 mod counter to generate 25 Mhz signal form the 50 MHZ system clock.

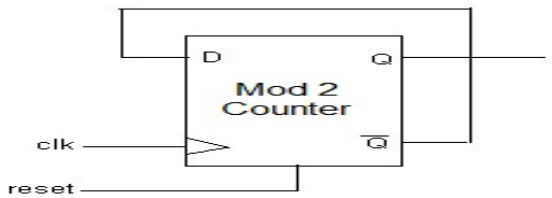

Figure-5. Mod counter.

We can design two individual counters. First counter counts from 0 to 639 to meet horizontal resolution, second counters counts from 0 to 479 to meet vertical resolution. We can also define two signals called h_end and v_end that indicates the point of the counters.

We can connect the VS and HS directly to the VGA connector. As we recalled the previous part, RGB (Red, Green and Blue) were the analog signals with ranging from 0 to 0, 7 volt. Hence we should use 270 ohm resistor between the RGB (Red, Green and Blue) and VGA connector pins. (inner resistor of the monitor is about 75 ohm and the output voltage of the FPGA is 3.3 Volt.)

We connect pin numbers 5, 6, 7, 8 and 10 to ground.

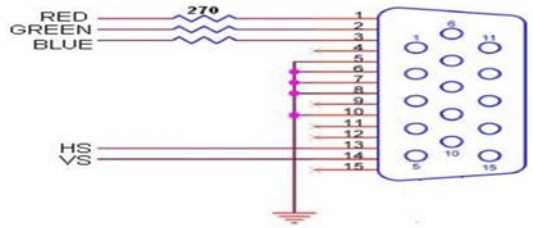

Figure-6. Connection with VGA connector.

Video_on signal value is one, then only when video is displayed on the monitor. HS and Video_on signal status will be as follows:

HS and Video_on signal status will be as follows:

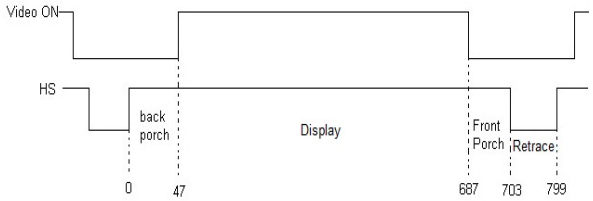

Figure-7. Horizontal signal.

VS and Video_on signal status will be as follows.

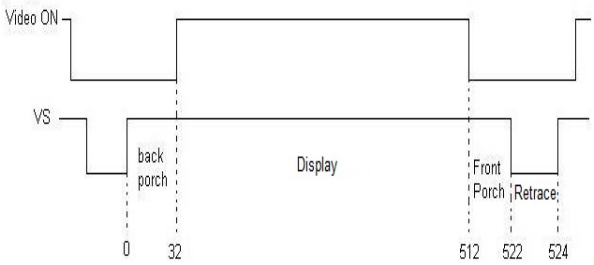

Figure-8. Vertical signal.

VGA SYNC MODULE CONFIGURATION:

In order to understand the VGA controller architecture and to control the LCD display in the single FPGA device.

Figure-9. Timing synchronization of VGA monitor display.

It contain 6 Sections
1) Hsync
2) Vsync
3) Vertical Front Porch
4) Vertical Back Porch
5) Horizontal Front Porch
6) Horizontal Back Porch

VERTICAL SYNCHRONIZATION

Display: In this regions the pixels are actually displayed on the screen.

Retrace: The time required for retracing the cursor to move back to the top-left corner when it reaches

the last pixel in the bottom right corner of the screen. In this duration the video color must be set to black.

Front Porch (porch before retrace): The blanking interval before the sync pulse is called as the front porch. Front porch forms the right border of the display region. In this region the video signal will be disabled.

Back Porch (porch after retrace): the blanking interval after the sync pulse is called as the back porch. Back porch forms the left border of the display region. In this region the video signal can be disabled.

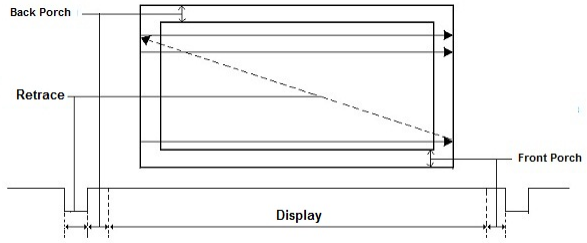

Figure-10. Vertical synchronization.

HORIZONTAL SYNCHRONIZATION:

Display: Region where the pixels are actually displayed on the screen.

Retrace: The time required for retracing the cursor to the beginning of the next row when it reaches the end of a row. This duration the video color must be set to black.

Front Porch (porch before retrace): The blanking interval before the sync pulse is known as the front porch. Front porch forms the right border of the display region. In this region the video signal should be disabled.

Back Porch (porch after retrace): The blanking interval after the sync pulse is known as the back porch. Back porch forms the left border of the display region. In this region the video signal should be disabled.

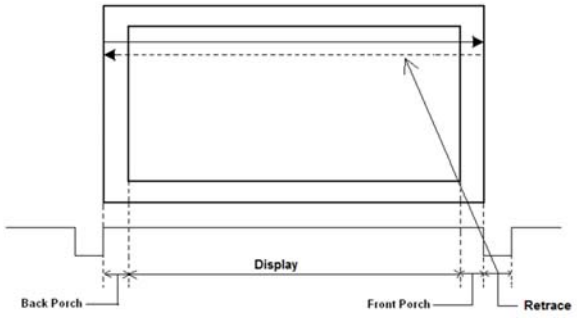

Figure-11. Horizontal synchronization.

IMAGE GENERATION MODULE

IMG GEN module will generate 3 bit RGB signal. It consists of three sub-modules. In this part we will design the objects shown on the LCD Monitor. In our project we will generate wall, bar and ball objects with animate the objects inside the LCD display screen

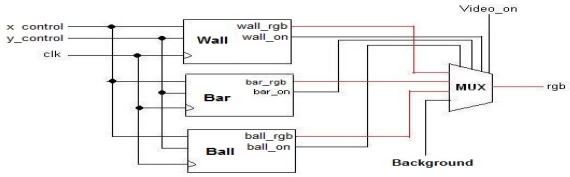

Figure-12. Image generation module.

HARDWARE CIRCUIT FOR VIDEO GAME:

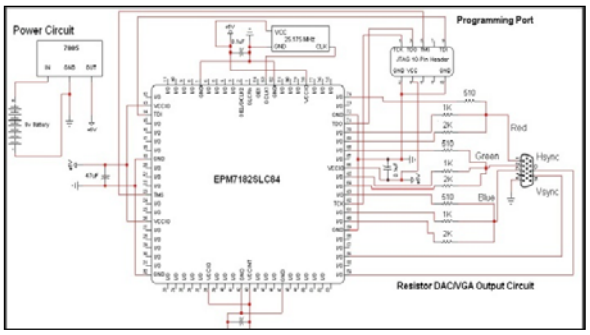

Figure-13. Circuit diagram.

Here we are using FPGA Kit, Crystal clock, VGA Connector. Hence initially image can be created and the animation can be given to the image.

This can be done by HDL Languages. By using VHDL language we are writing the program to create image and once the image has been generated, the animation program will be written.

The wall, a ball and a bar is the main components in this game. These components will be created by using the program and also we are initializing the positions of these components.

The below Figure-12 shows the hardware kit for this video creation.

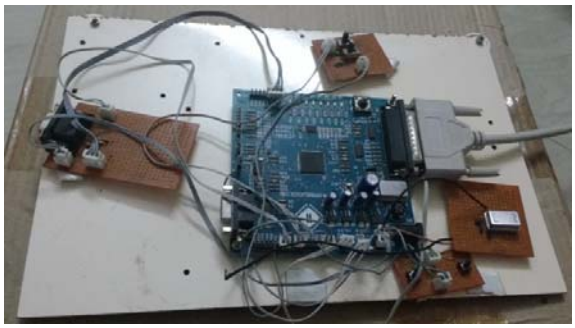

Figure-14. FPGA kit for video game.

We are using Switches to control the movement of the bar. Finally we can see the game in LCD monitor to play the game. The background colour, wall, ball, bar colours can be changed within the program.

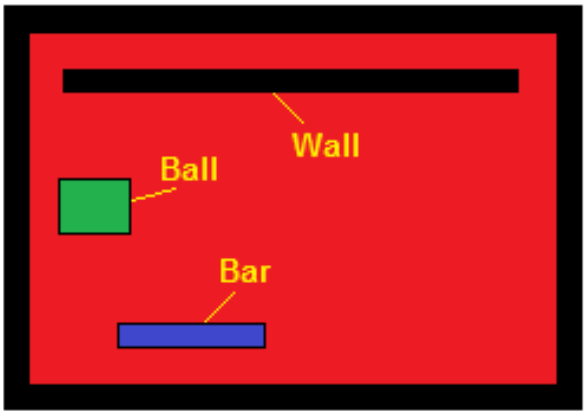

Figure-15. Video game output.

FUTURE WORK

By using this concept we can create videos with standard image. This can be used in various fields in ATM machine, Notice board system etc. We can go for higher level of game creation in future.

REFERENCES

[1] Hungary D. Vincze; S. Kovács. T. Tompa Department of Information Technology, University of Miskolc, The Pong game implementation with the FRIQ-learning reinforcement learning algorithm.

[2] Gao. X, Dept. of Comput. Sci., New Jersey Inst. of Technol., Newark, NJ, USA. Tian. J; Liang. X; Wang. G, ARPP: An Augmented Reality 3D ping-pong game system on Android mobile platform.

[3] Martin Werner, Benno Stabernack, Christian Riechert, Hardware Implementation of a Full HD Real Time Disparity Estimation Algorithm.

[4] Riechert, F. Zilly, M. Muller and P. Kauff. 2012. Advanced interpolation filters for depth image based rendering. in Proceedings of the IEEE3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video, Zurich, pp. 1-4.

[5] Riechert, F. Zilly, P. Kauff, J. Guether, and R. Schaefer. 2012. Fully automatic stereo-to-multiview conversion in autosteroscopic displays. The Best of IET and IBC. 4: 8-14.

[6] România A. Gontean, R. Szabó Politehnica University Timişoara, Faculty of Electronics and Telecommunications. Pong game on FPGA with CRT or LCD display and push button controls.

[7] Zhang. L, Zhang. K, Chang T.S, Lafruit.G, Kuzmanov. G. K and Verkest. D. 2011. Real-time high-definition stereo matching on FPGA. In: Proceedings of the 19th ACM/SIGDA International Symposium on Field Programmable Gate Arrays, New York, NY, USA. pp. 55-64.