The University of Danang

Danang University of Technology

Electronic and Communication Engineering

--o0o--

EE 271_Final Project

MARIO GAME

M.A.D Team - Group 13 - 09ECE

Nguyen Ngoc Minh
Ngo Tran Duc Thang
Nguyen Xuan Tien

Table of Contents

Introduction

Our project is about creating Mario game using KIT DE1 of Altera. We intended to reproduce some parts of the very famous game; that is Mario. Besides, we also created some features for this game based on our creativity to make the game more interesting and different in compare with the old version.

Objectives and overview

The purpose of this lab is to bring many of the things we've been studying this term together into a final project. We want to learn more about the process of designing and developing digital systems and to use our imagination and creativity as we solve real world engineering problems.

For this project, our goals will be to design, develop, and implement a simple digital system of our own design.

Project

I)   Design Specification.

- Control MARIO moving up, down, left, right to overcome in turn obstacles such as monsters, pipes, bricks, and dragons to reach the final destination - rescue the princess.
- During his journey, there are some cases that he could meet:
  - Mario touches two sides of the monster (goompa) => Mario dies.
  - Mario is on the top of the monster => monster dies.
  - Mario touches two sides or the bottom of the dragon=> Mario dies.
  - Mario is on the top of the dragon => dragon dies.
  - Mario eats mushroom => its size is double.
  - Mario's head butts the bottom of the brick => mushroom appears.
  - Mario falls down the hole => Game is restarted.
  - Mario kills dragon => the Princess appears and win the game.
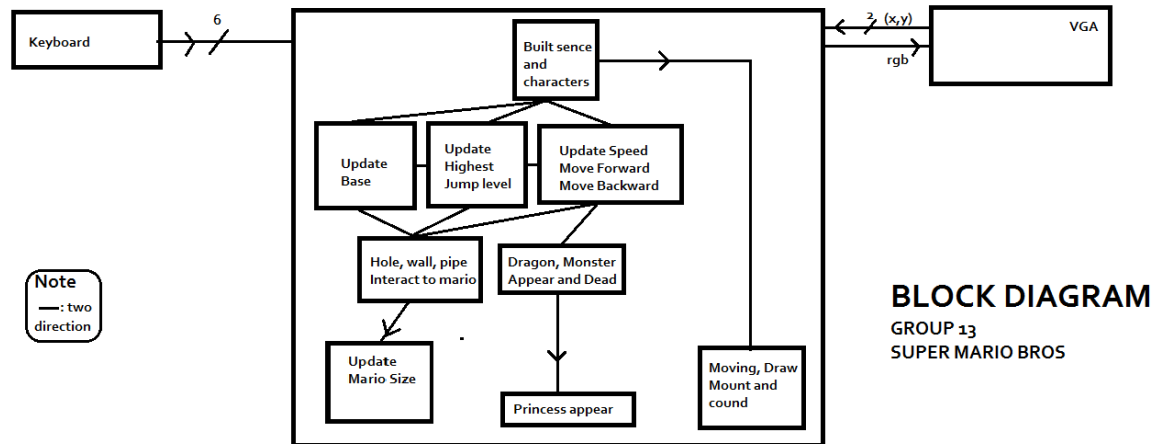
II)   Design Description.

1. Hardware Configuration

- Monitor for showing our result.
- Keyboard as input to control the game.

2. Software Configuration

- Notepad ++ for coding.
- QuartusII and the Altera Cyclone II FPGA for compiling.
- Ascgen2-2.0.0 for converting rgb or black-white images to binary ones.

-

III) Block and State Diagram.

BLOCK DIAGRAM
GROUP 13
SUPER MARIO BROS
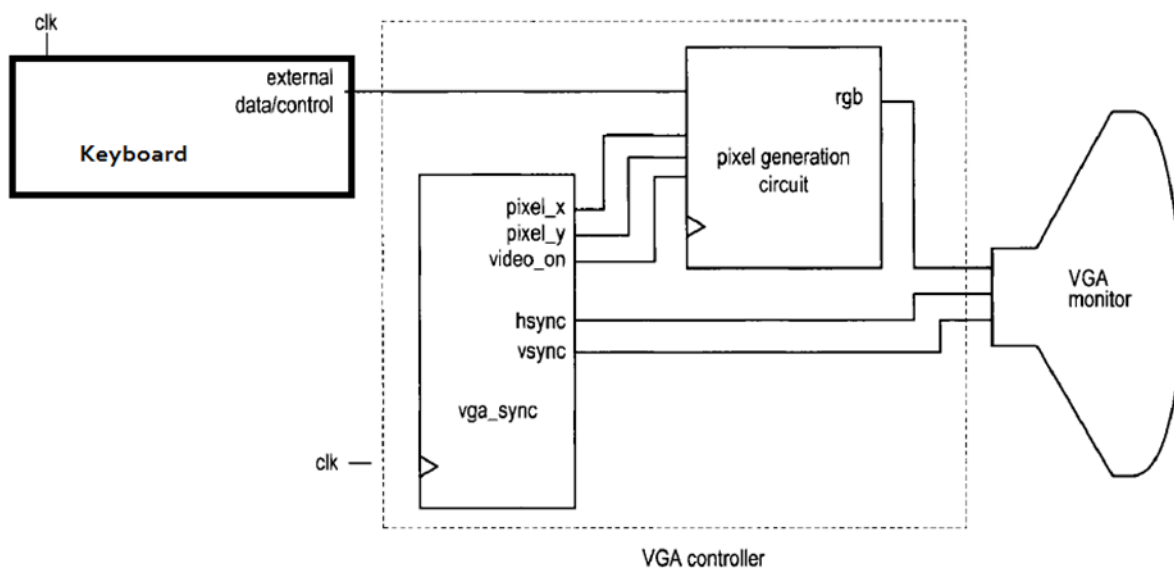
IV) I/O Modules: VGA, PS2 Keyboard.
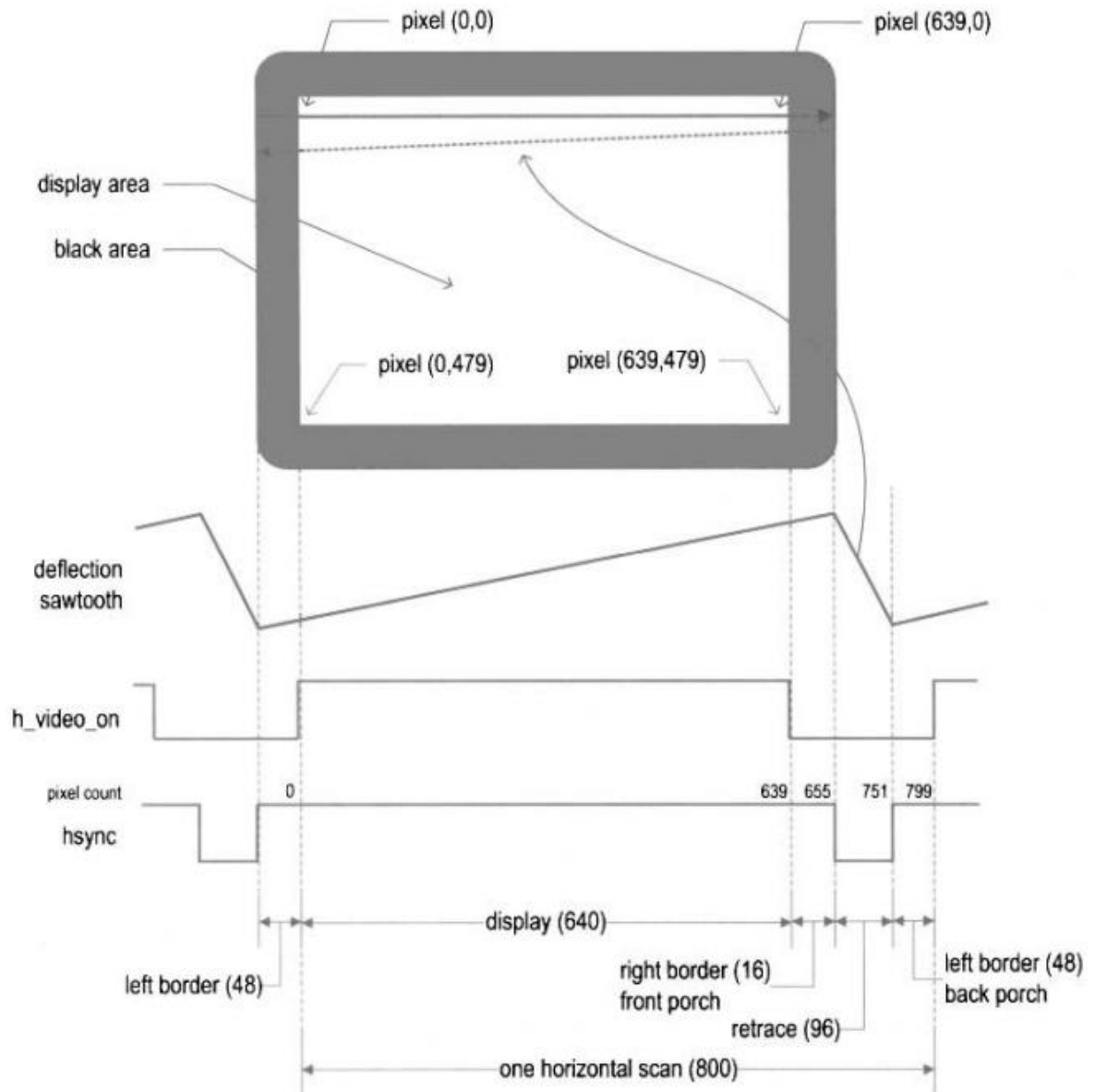
1) VGA Controller: Graphic

- We used 9-bit color 640-by-480 resolution interface for LCD.
- Hereunder is the simplified block diagram of a VGA controller.
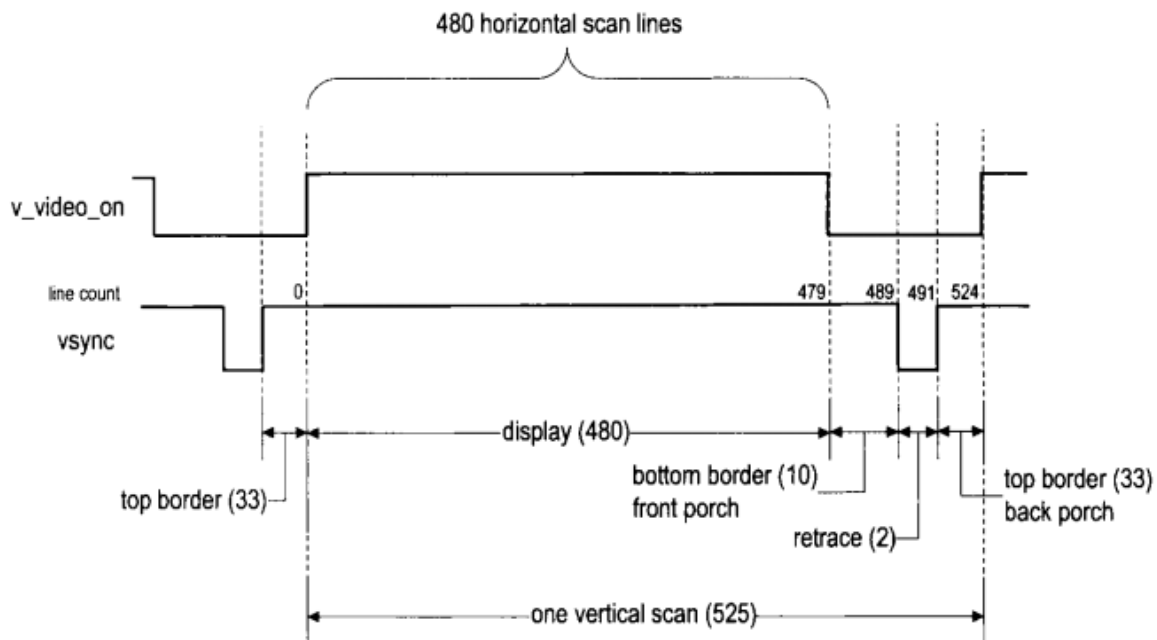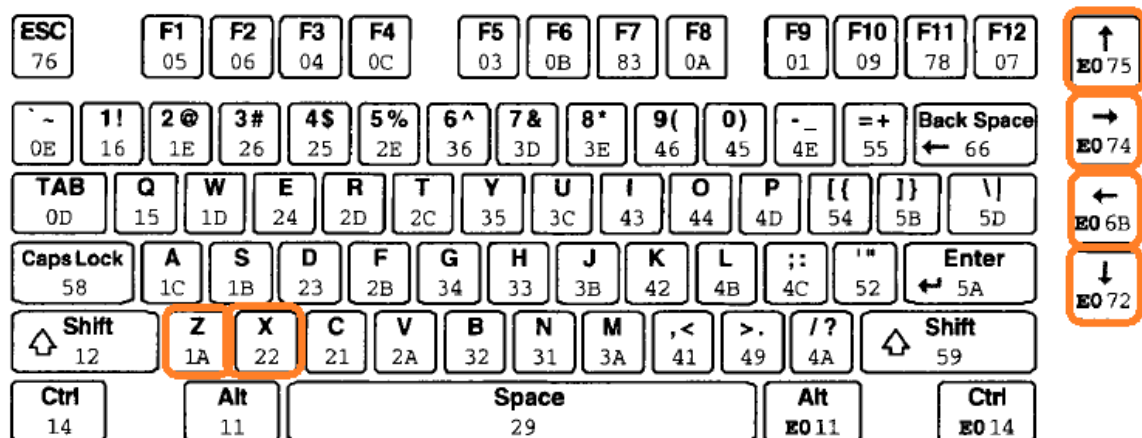

- The VGA synchronization

+ Horizontal synchronization.

+ Vertical synchronization

2) PS2 Keyboard

- Keyboard receives PS2_CLK and PS2_DATA from kit DE1 to create six signals control behavior of Mario character.

- The keyboard procedure 8 bits for each time player press button. In this project we only use 6 buttons. They are up, down, left, right buttons and "Z" for inceasing speed and "X" for jumping.
- Module keyboard in Mario_top.v is written for this purpose.

V)   Working Plan.

VI) TestCases, TestBench and Simulation in ModelSim.

1. Run back and forth.
   - Mario cannot move (or simply cannot go through) if right behind or right front of him is any obstacle.
   - Mario can move back and forth easily if there is no obstacles behind or in front of him.

2. Jump up and down
   - Mario jumps up to a maximum high and drops.
   - He will wait until reaching ground to jump again.
   - When he drops, nothing can make him up until he reaches the base and "jump" button to be pressed again.
   - If he is under the brick, the maximum high that he can jump is the brick's bottom.

3. Shift background

- When the Mario reaches the middle of the screen, the background and object will shift to the left. This makes illusion that Mario is still moving forward.

4. Interact with wall

- Mario can move back or forth and jump easily if he is standing in the wall.
- The maximum high Mario can reach is the wall's bottom.
- The Mario cannot go through the left or right edges of the wall.

5. Interact with pipe

- Similar to the wall, except that Mario cannot be under the pipe. The reason is that the pipe in on the ground while the wall is higher than the ground 150 pixels.

6. Draw Mario, cloud, and wall in full color

- Understand and be able to draw object in $2^9 = 512$ colors.

7. Interact with monster and dragon

- Mario touches two sides of the monster (goompa) => Mario dies.
- Mario is on the top of the monster => monster dies.
- Mario touches two sides or the bottom of the dragon=> Mario dies.
- Mario is on the top of the dragon => dragon dies.

8. Interact with mushroom

- Mario butts at the exactly the position of mushroom inside the brick => mushroom appears.
- Mario eats mushroom => his size is double.
- Path of the mushroom: appear from the wall brick -> move right -> fall down the ground -> move right until touch the pipe -> move left.

9. Interact with hole

- Mario falls into the hole => game is restarted.

10. Interact with the Princess

- When Mario kills the dragon; immediately, the Princess appears.

VII) Comments and Discussion.

1) What we did.

- Finished the Mario's moving module, which is the most difficult part of our project; including moving to the left, right with different speeds; jumping up and down with exact angle (without crossing edges of the

brick, going through the pipe or hole,…); jumping and moving to hide and to kill the monster as well as the dragon.

- Building scenes, objects and characters for the game, including cloud, pipe, brick, mountain, monster, princess, and especially big and small-size MARIO.

2) What we haven't finished yet.

- Mario goes through the pipe to move to other scenes.
- Text introduction.
- The sound system.

VIII) Development.

- Make the differences between each levels of the game.
- Create the text introduction and guide.
- Complete the sound system for this game.
- Built more scene

Reference.

1) *FPGA Prototyping by Verilog Examples: Xilinx Spartan- 3Version _PONG P.CHU (http://academic.csuohio.edu/chu_p/rtl/index.html)*
2) *DE1 User manual*
3) *Create binary picture http://www.glassgiant.com/ascii/*
4) *http://www.scribd.com/doc/48288108/Rapid-Prototyping-of-Digital-Systems-SOPC-Edition*
5) *Icon model http://www.iconexperience.com/m_collection/search/?q=tree*
6) *Audio Codec http://www.fpga4fun.com/*
7) *Inspiration for Mario game (http://vimeo.com/9928343)*
8) *Color using 9 bits https://letshackstuff.com/?s=blog&m=tag&x=fpga*