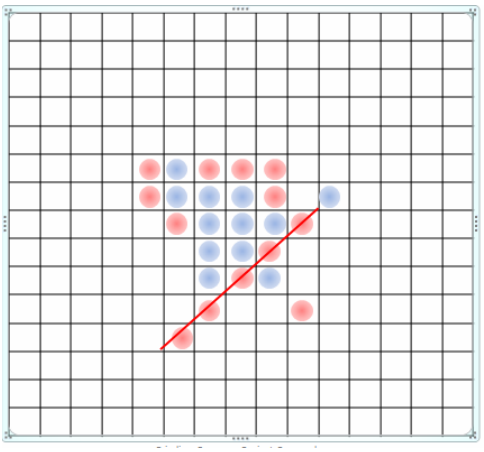
LUND UNIVERSITY

EDA385 The Embedded System Design - Advanced Course

5-IN-LINE (GOMOKU) BOARD GAMES

PROJECT SUPERVISOR: Flavius Gruian & Per Andersson



Date: ***20010-09-05***

SUBMITTED BY: ***Zhonghua Wang(mas09wz1@student.lth.se),***
Ziyang Li (mas0zl1@student.lth.se),
Hongwan Qin (mas09hq0@student.lth.se)

Lund 2010

5-IN-LINE(GOMOKU) BOARD GAMES

Abstract

This project report about 5-in-line board games by using the hardware platforms as Digilent NEXYS-2 boards, AT PS/2 keyboard, and 640x480@60MHz VGA to implement on Xilinx's EDK(Embedded Development Kit) as design environment. This project includes three important parts : VGA controller, PS/2 controller, and Main program in MicroBlaze. This design involves the software and the hardware cooperation though the setting on EDK.

1 Introduction

In this board game design, we firstly should introduce the design tools specification. For our project, there are one FPGA board- Digilent Nexys2 Board based on a Xilinx Spartan 3E FPGA [1], one generic VGA to display the image (the game area) which manage 640x480 resolutions, AT PS/2 Keyboard to send the player's data into system.

1.1 FPGA Board (Digilent Nexys2 Board) Specification

This board as figure 1 has high-speed USB2 port, 16Mbytes of RAM and ROM, and several I/O devices and ports for project design, the embedded processor systems based on Xilinx's MicroBlaze (the BRAM size16KB) so that it could fully run on Xilinx ISE tools.[1]. The Nexys2 board frequency is **50MHz**. In this board there are PS/2 Port can accommodate the PS/2 mouse or keyboard, VGA port with 8-bit color and the two standard sync signals (HS- Horizontal Sync, and VS- Vertical Sync). Those on-board ports as Fig.2.

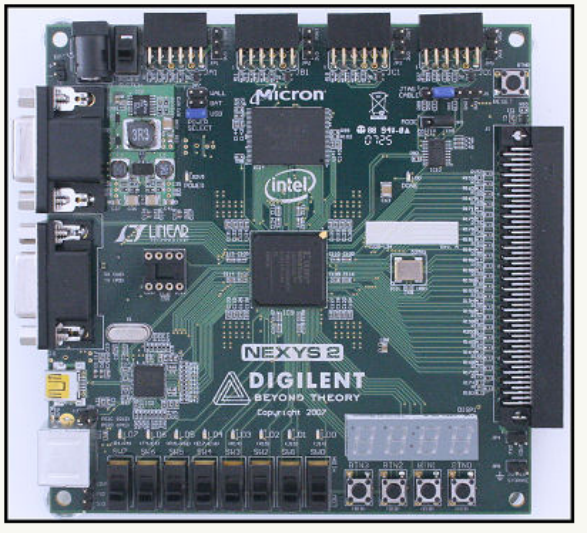


Figure 1: Digilent Nexys2 Board

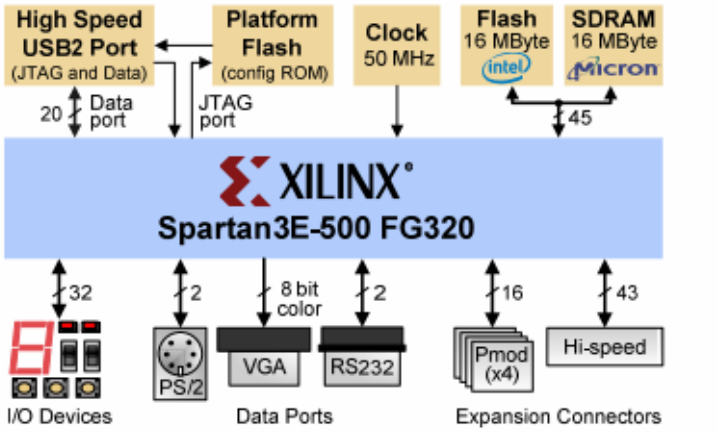


Figure 2: Nexys2 Board Circuit Structure

1.2 The AT PS/2Keyboards Specification

The PS/2 keyboard is connected to the external equipment to send data to the host and the host can send data to the keyboard. The data length is 11 bits as Fig.3 with one start bit (logic 0), 8 data bits (lsb first), odd parity bit and a stop bit (logic 1). The clock rate is approximately 10 to 30 kHz and varies from keyboard to keyboard [2].

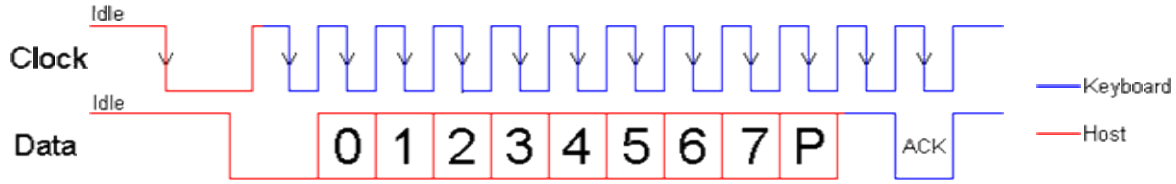


Figure 3: PS/2 Keyboard Protocol

Below diagram as Fig.4 shows the Scan Code assigned to the individual keys. We will choose some of scan code as our controller for chess moving.

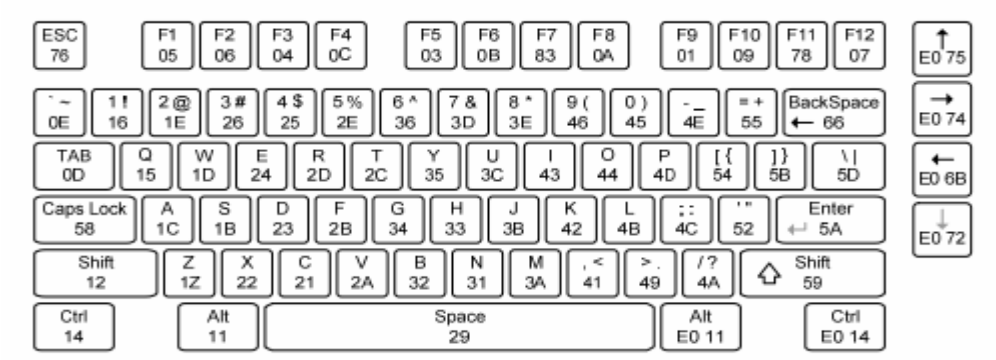


Figure 4 : PS/2 Keyboard Scan Codes

1.3 VGA Specification

The VGA display with pixel resolution of 640×480@60Hz is used manages the 640x480 resollution,which the VGA controller is designed based on a system clock of **25MHz** which divides the input clock frequency by a factor 2 as Fig 5. The VGA signal timing diagram is shown in below Fig.6.

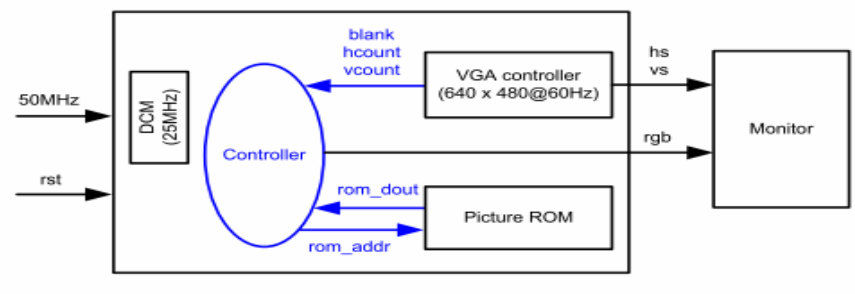


Figure 5: VGA Controller Design Diagram

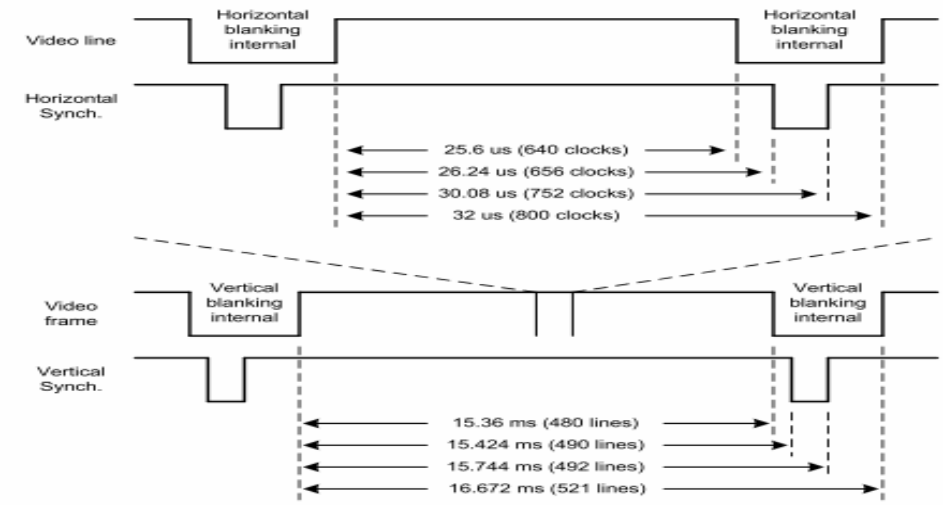


Figure 6: Signal timing diagram for VGA display

1.4 Architecture

Below figure presents the preliminary architecture of our 5 in line game.

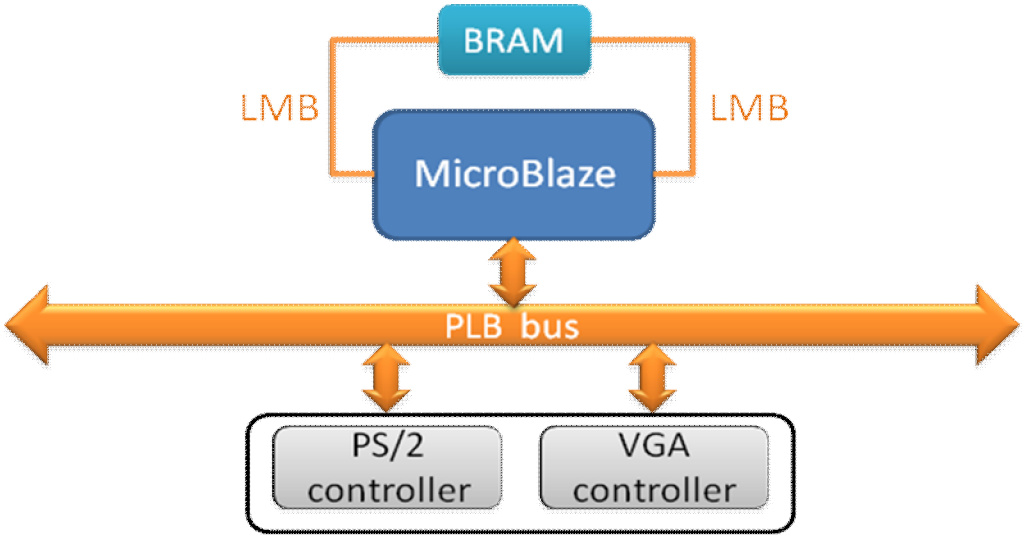


Figure 7: Preliminary Architecture

2 Game Design Introduction

In this section, we will give a draft explain about our design idea.

2.1 Basic Game Design Structure

Five in line game on the 10×10 boards the abstract strategy board game also called Gomoku. It is traditionally played with go pieces (black and white stones) Black plays first, and players alternate in placing a stone of their color on an empty intersection. The winner is the first player to get an unbroken row of five stones horizontally,

vertically, or diagonally.[4]

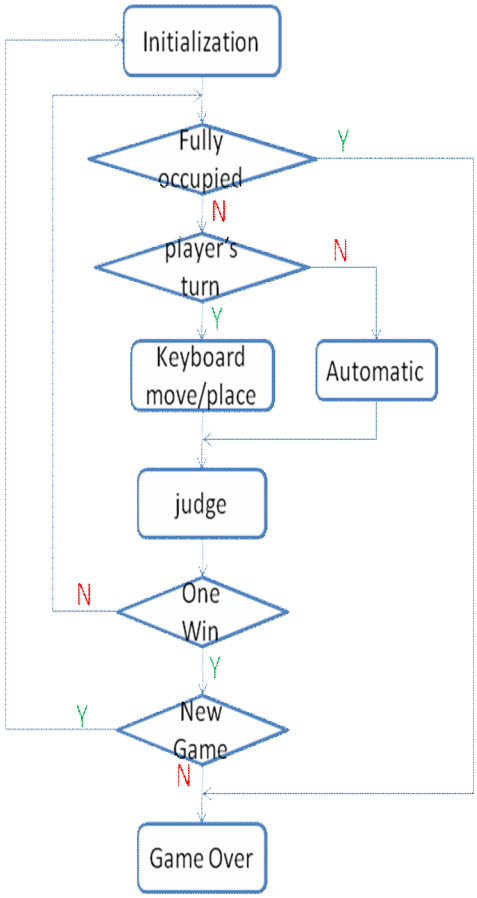


Figure : 5_In_Line Procedure Flow Chart

As you see, we have defined this 5 in line board game flow chart; there are nine basic modules in this design.

(1) Initialization: Firstly, create game board matrix data, reset the opponents' data to zero; then initial the I/O of VGA, PS/2 AT key board, and draw out the game area in VGA.

(2) Fully Occupied: This module will check whether the whole play area is full of chess or not. If it is full occupied, then the game should be over; if not, the program goes to the next step.

(3) Player's turn: This module works as coordinator to switch the play sequence.

(4) Keyboard move/place: Though keyboard to choose the place for chess. In this step we have to decide is the selected place from the key board available.

(5)Automatic: According to some algorithm to estimate the player's status of chessboard and then computer places chess, which choose the place for chess without keyboard.

(6)Judge: According to the rule to check whether 5 in line, if yes, the one will win, if no, game continues.

(7) New game: Start the new game.

(8)Game Over: End this game.

2.2 Computation complexity

Currently, three problems as below are relatively complex:

1) Algorithm for the rival's (machine) decision
2) Judgment of the Victory/Defeat
3) Pixel calculation for the VGA displaying

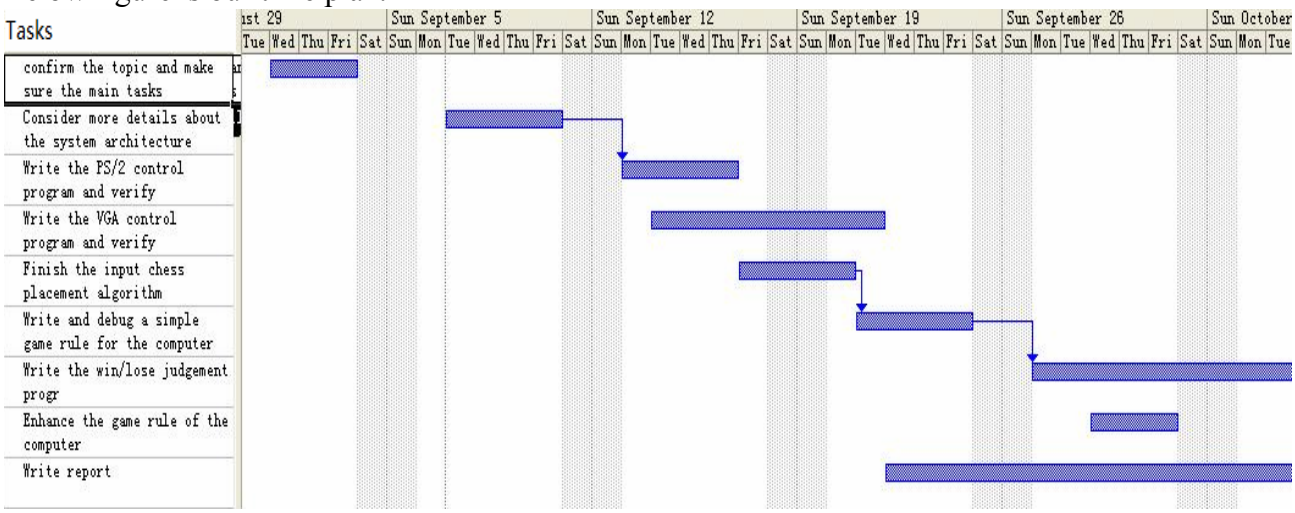
The first problem is that we have to design a reasonable algorithm for the machine to figure out which place should it lay its chess to. That means we should design a computer program of automatic calculating and running for the machine. This task may be stiff because it is hard for us to design a complex algorithm for the machine in a short time. So we plan to design a relative simple algorithm firstly and then try to find out a better method when the basic function is being carried out.

So far we have made out a method for judgment of the victory or defeat.

Figuring out how to display different color chess in the game board is also a rough job.

2.3 Time Plan

Below figure is our time plan.



References

[1] Digilent Nexys2 Board Reference Manual, Revision : June 21,2008
[2] http://www.beyondlogic.org/keyboard/keybrd.htm
[3] VGA Component Reference Design , Revision: April 17, 2007
[4] http://en.wikipedia.org/wiki/Gomoku
[3] XILINX, “ISE Quick Start Tutorial”. 1-800-255-7778