American Pool Video Game

Group Name: Pool-Maniac

Jiawan Zhang jz2492

Xunchi Wu xw2256

Yuhan Zhang yz2500

Zeshi Wang zw2221

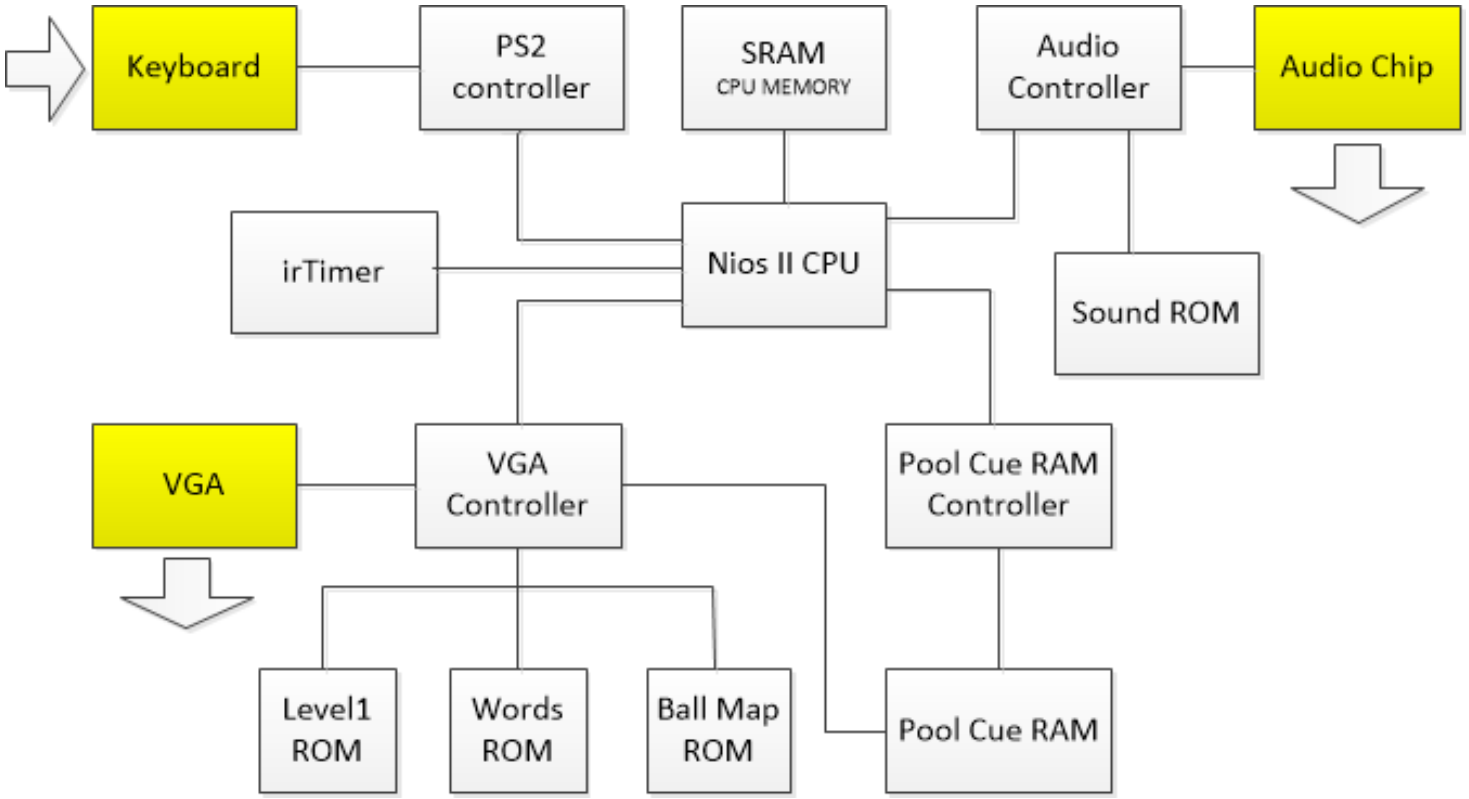
Yichen Liu yl2904

CSEE 4840 Embedded System Design

Overview of the project

- Classic American Pool Game
- VGA display
- Keyboard for customer's inputs
- Audio play sounds when collide
- Software suitable for FPGA

Architecture



Who Did What?

- Jiawan Zhang: VGA, Software
- Xunchi Wu: Software
- Yichen Liu: Network
- Yuhan Zhang: Software
- Zeshi Wang: Keyboard, Audio

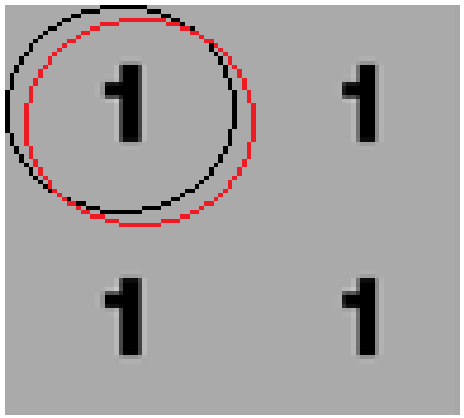
VGA Controller

The Video display part has 5 sprites (Level 0 to Level 4) concluded as following:

	Contents
Level 0	a. Frames of spool table and serve line b. Strength bar c. Background
Level 1	a. 6 pockets and around areas
Level 2	a. 16 balls
Level 3	a. Pool cue
Level 4	a. Instruction words

VGA Controller

- **Balls' Rotation**
- Use Map to make ball rotate



- When the ball moves, we move its bias to the map with an opposite direction。

VGA Controller

- **Pool Cue**
- Use 31 images for pool cue. Use mirror to make 120 states.
- Use Only 2*355 16-bit data for one image, which gives only the start and end positions of the cue in each line of the image.
- Build an RAM with on chip memory to let the VGA controller use the data in software.

PS2 Keyboard

- The keyboard is the only input for the pool video game.

- **Function for each key**:
- Arrow keys**:** used to move the pool cue and adjust the position of cue ball.
- Space key**:** used to adjust the strength and place the cue ball.
- Enter key: used to start the game and hit the ball.

- Hardware is implemented with the **PS2 controller** based on lab 3.
- Build the "**keyboard.h**" file in the software for keyboard.

Audio

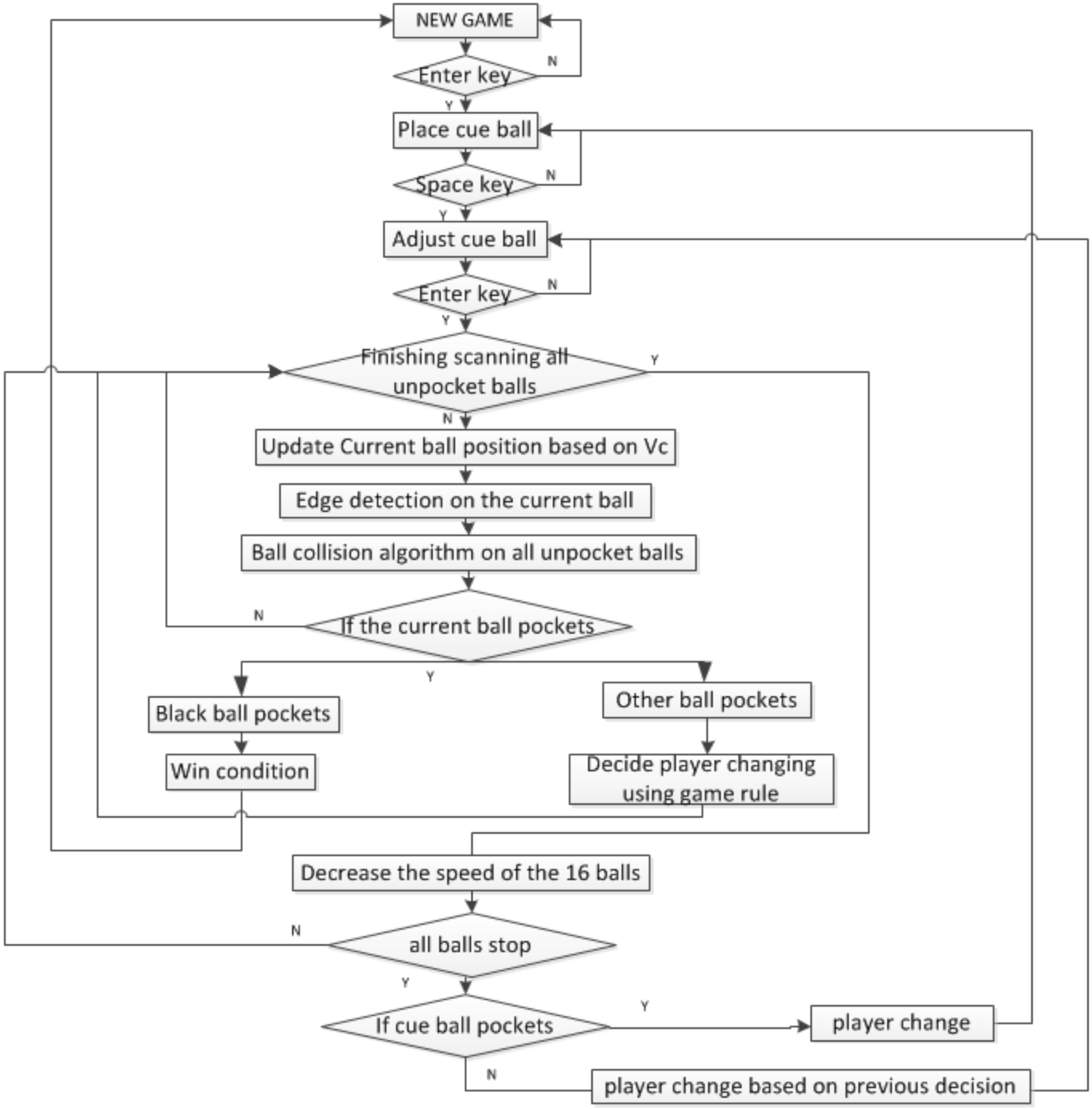
- **The audio block can generate sounds for collisions**

- sounds: volume will change according to the speed of the ball.
- **Hardware**: Store the audio information in the on-chip ROM. Triggered with a start signal settling by the writedata.
- **Software:** Choose a sound and write to hardware. according to the speed and collision type of the balls.

SOFTWARE AND ALGORITHM

- The overall game logic control
- The collision algorithm between balls and the moving parameters afterwards
- The collision algorithm between ball and table edges and the moving parameters afterwards.
- The pocketing of the balls and player changing logic.

FLOW CHART


NEW GAME
Enter key
N
Y
Place cue ball
Space key
N
Y
Adjust cue ball
Enter key
N
Y
Finishing scanning all unpocket balls
Y
N
Update Current ball position based on Vc
Edge detection on the current ball
Ball collision algorithm on all unpocket balls
If the current ball pockets
N
Y
Black ball pockets
Win condition
Other ball pockets
Decide player changing using game rule
Decrease the speed of the 16 balls
all balls stop
N
Y
If cue ball pockets
Y
player change
N
player change based on previous decision

Network

- Partially Finished:
- Established the communication between DE2 board and laptop via UDP protocol using Python programming.
- Finished the GUI design on laptop using JAVA programming.

CHALLENGES

- VGA: Balls and pool cue's rotation.
- Software: Working with low "CPU" speed and low display resolution.

SUMMARY AND WHAT WE LEARNT

- We made our game work as promised except the network part.

- What we learnt except the knowledge of Embedded System.
- Never under-estimate the task you undertake, because things will always be easier said than done.
- Starting early and trying to finish every possible part as soon as you can will always help in such a big and complicated project like this.
- Because you will find a lot of bugs and mismatching incidents later when you try to combine all the parts together.

Enjoy the Game !