Virtual Basketball: How Well Do You Shoot?

Chun Li & Jingwen Ouyang

6.111 Spring 2007
System Overview

- ADXL330 3-Axis Accelerometer
  - analog x 10
  - analog y 10
  - analog z 10

- Calibration & Velocity Calculation
  - velocity x 10
  - velocity y 10
  - velocity z 10
  - position x 10
  - position y 10
  - position z 10

- Game Display
  - vga_colors 24
  - vsync, hsync, hblank

- Chun
- Jingwen
Accelerometer & A/D Converter

• ADXL330 3-axis Accelerometer
  • ±3g range
  • Operating voltage 3V
  • 300 mV/g sensitivity

• AD7810 10-bit A/D Converter
  • Sample at 1 kHz

Figure 16. AD7810 Serial Interface Timing
Calibration

- Obtain player height
  - Calculate initial x, y, z positions

- Why: Variances
  - 0g bias point
  - Sensitivity
Game Display Overview

Initial positions
10 x 3

Initial speeds
10 x 3

labkit_clock

DCM

pixel_clock

To All

Set

Debouncer/ Synchronizer

reset_sync

replay_sync

Game Logic

Ball x, y, z Positions
10 x 3

Shadow Positions
10 x 3

pixel_count 10

line_count 10

VGA Controller

Display Field

1 x 4 VGA signals

24 rgb_signal
Get Ball position (FSM)

Get Shadow Position

Game Logic

Initial Positions

Initial speeds
Calculations

• **Speed**
  - \( V_z = g \Delta t + V_z \) (only vertical direction has acceleration)
  - \( \Delta t \) = time between each frame

• **Position**
  - \( X = V_x \Delta t + X \)
  - \( Y = V_y \Delta t + Y \)
  - \( X = V_z \Delta t + Z \)

• **3D to 2D conversion**
  - \( W = Z + X \sin(a) - Y \sin(b) \)
  - \( U = X \cos(a) + Y \cos(b) \)
Display Field

3D to 2D Converter

ball_x, y

Address Counter
For ROM

pixel_count 10
line_count 10

7 ball_addr

ROM

2 color_index

Color Mapping

24 rgb_ball

RGB Controller

24 rgb_signal

rgb signals for the shadow, player, scores, and background, which are obtained similarly
Game Display Interface (1)

+ basketball
+ basketball
+ 2
+ 4
+ score

Player shooting basketball.
Game Display Interface (2)
Conclusion

- practice shooting without a court
- Whenever
- Game for the Wii