FPGA Hunt

Shooting the dog... Priceless
FPGA Hunt Overview

- Duck Hunt 2.0
  - More Creature Types
  - Dynamic Gaming Experience
  - Shoot the Annoying Dog
  - Improved Graphics
  - Functions on any screen (LCD/CRT...)

Zapper Functionality

- Uses LED's around LCD to determine direction
- Connects to PC via USB
- Will Utilize Logic Analyzer to Help Integrate
Zapper Logic

- Informs Game Logic Unit when Trigger Pressed
- Translates coordinates from Zapper to GLU
- Calibrates Zapper for Any Screen Size
Game Logic Module

- Inputs
  - Triggered
  - (x, y)

- Outputs:
  - Object type
  - Object Position
  - Object Status
  - Game Time
  - Player Score
  - Player Health / Ammo (If Time Permits)
Game Logic Internal Workings

- Finite State Machine
  - Updates every $1/60^{th}$ of a second
  - Each state correlates to a scene/level
  - At beginning of each level, spawn enemies
  - Upon trigger, test position for enemy collision
    - Vary fuzziness related to current difficulty level
  - Upon certain events (enemy destroyed):
    - Update Score, Change State
Graphics Overview

- Objects read from Environment Lists
- Graphics farmed out to submodules
- Pixels rendered in order of depth
Graphics Main Module

- Driven by pixel clock
- Asks each object for a pixel and its depth
- Picks pixel based on depth comparison
Environment Lists

- Decouples Dependency on GLU
- Encodes Obj Imformation in Registers
- Generalizes The Objects on Screen

Obj 1
Obj 2
Obj 3.
.  
.  

Graphics Modules...
What is a an Object?

- Animated Objects – Goose, Dog
- Background Objs – Trees, Clouds
- Each is a module
- Will maintain state (unrelated to GLU)
Graphic Modularity

- Decoupling Leads to Freedom
  - Once Basic Infrastructure in place...
  - Expansions easy
- Keeping extra state in objects
  - More complex animations
My One True Fear

8 Shootable Objs + 8 Background Objs = 16 Objects

n Sprites per Object = 16n

Assume a max 200x200 pixels per obj:

40,000 pixels * 16n = 640,000*n pixels

640,000 * 3 bits = 234*n kilobytes

We have 4 MB ram...

4 MB / 234 kB = ~17 sprites / obj !!!
## Development Time Line

- **Four Milestones**

<table>
<thead>
<tr>
<th>Milestone</th>
<th><strong>Game Logic</strong></th>
<th><strong>Graphics</strong></th>
<th><strong>Zapper/Sound</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Module</strong></td>
<td><strong>Peak</strong></td>
<td><strong>Pete</strong></td>
<td><strong>Farreed</strong></td>
</tr>
<tr>
<td>Monday, Nov 19th</td>
<td>Interface to Zapper</td>
<td>1 Sprite fully functional (Including fly/die)</td>
<td>Obtain Zapper from Hong Kong</td>
</tr>
<tr>
<td></td>
<td>Make one Goose Move</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monday, Nov 26th</td>
<td>Write multiple objects to environment list</td>
<td>Read object from env list</td>
<td>Determine pin out Output to GLU on trigger</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Finish Background</td>
<td></td>
</tr>
<tr>
<td>Monday, Dec 3rd</td>
<td>Test integration with Graphics and Zapper</td>
<td>Full integration w/ Game Logic</td>
<td>Zapper calibration for LCD Position output to GL</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monday, Dec 10th</td>
<td>Testing/debug completed</td>
<td>Testing complete More advanced graphics</td>
<td>Work on sound (if time allows)</td>
</tr>
</tbody>
</table>
Acknowledgements

- Owlo.com – Green Feathers Image
- Nintendo – DuckHunt Images