Donkey Kong

Sean Stothers (sps2308), Ania Róża Krzyżańska (ark2219), Ines Khouider (ik2512)
Goal

Make the arcade game Donkey Kong
Mario jumps over barrels
Mario dies when a barrel hits him
Background Tiles

Diagram:

- VGA Ball
- RAM (Array Tiles)
- RAM (Pixel Map)
- Color Table

Connections:
- H count V count
- Tile Type
- Color Num
Color Table and Tile Map

Array of 1024 different addresses (32x32). Each address corresponds to an 8x8 tile.
Pixel Map
Sprites
Game Logic

- **Reset Game**
  - Mario in start
  - Barrels Hidden
  - Heart Hidden
  - Score = 0

- **Button Pressed**
  - Left/Right/Jump with
  - grider
  - up/down near ladder

- **Sprite Location**

- **Updates Mario**
  - Mario Falling?
  - Won?
  - Snap up?
  - Dead?

- **Barrel Update**
  - For each barrel, move if not hidden.

- **Win Game**

- **Lose Game**

- **Run Game**

- **Is Jumping?**
## Resource Budgets

<table>
<thead>
<tr>
<th>Category</th>
<th>Size (bits)</th>
<th>Number of images</th>
<th>Total Size (bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mario</td>
<td>32 x 32</td>
<td>16</td>
<td>3x32x32x16 = 49152</td>
</tr>
<tr>
<td>Donkey Kong</td>
<td>50x50</td>
<td>1</td>
<td>3x50x50x1 = 7500</td>
</tr>
<tr>
<td>Barrel</td>
<td>16x16</td>
<td>2</td>
<td>3x16x16x2 = 1536</td>
</tr>
<tr>
<td>Ladders</td>
<td>20x30</td>
<td>9</td>
<td>3x20x30x9 = 16200</td>
</tr>
<tr>
<td>Platform</td>
<td>20x30</td>
<td>15</td>
<td>3x20x30x15 = 27000</td>
</tr>
<tr>
<td>Hammer</td>
<td>10x10</td>
<td>1</td>
<td>3x10x10x1 = 300</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td>101, 688 bits</td>
</tr>
</tbody>
</table>
Hardware Software Interface

- Game displayed on computer monitor
- We used 10 32-bit registers
  - Registers for tile map and array & mario coordinates
  - Separate register for each barrel
  - Joint register for sprite states
- NES controller protocol:

<table>
<thead>
<tr>
<th>Nothing:</th>
<th>01 7f 7f 7f 7f 0f 00 00</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left:</td>
<td>01 7f 7f 00 7f 0f 00 00</td>
</tr>
<tr>
<td>Right:</td>
<td>01 7f 7f ff 7f 0f 00 00</td>
</tr>
</tbody>
</table>

Direction
States
## Registers

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Sprite X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Sprite X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Sprite Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Sprite Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>L/R Mario</td>
<td>Which Mario</td>
<td>Which b0</td>
<td>b0 flip?</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Which b1</td>
<td>b1 flip?</td>
<td>Which b2</td>
<td>b2 flip?</td>
<td>Which b3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>b3 Flip?</td>
<td>Which b4</td>
<td>b4 flip?</td>
<td>which b5</td>
<td>b4 flip?</td>
<td>b0 on</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>b1 on</td>
<td>b2 on</td>
<td>b3 on</td>
<td>b4 on</td>
<td>b5 on</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Challenges

- Consistent hidden barrels → created bitmask
  
  \[
  \text{st.state} = (\text{sprite.tag} \ll 5) + (\text{sprite.direction} \ll 7) + (\text{st.state} \& 4294967071); \ //\text{this is the value of the bitmask with 32 1s and 0s at bits 5-7}
  \]

- Lining up mario to the slanted Girder → modified tilemap
Demonstration!