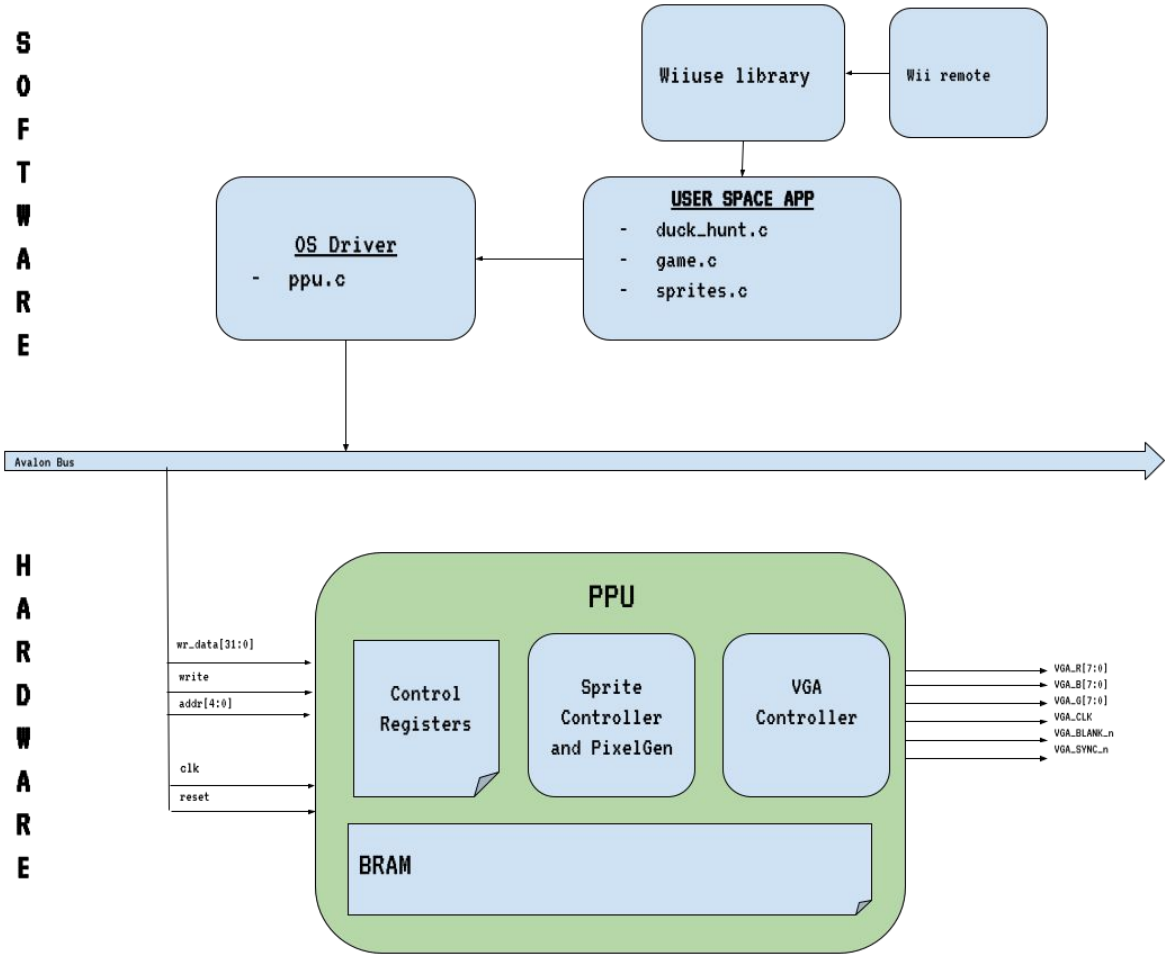

DUCK HUNT '22
Created by:
Alex Yao (awy2108), Bryce Natter (bdn2113),
and Kristen Shaker (kls2243)


SOFTWARE
Wiiuse library
Wii remote
USER SPACE APP
- duck_hunt.c
- game.c
- sprites.c
OS Driver
- ppu.c
Avalon Bus
HARDWARE
PPU
wr_data[31:0]
write
addr[4:0]
clk
reset
Control Registers
Sprite Controller and PixelGen
VGA Controller
BRAM
VGA_R[7:0]
VGA_B[7:0]
VGA_G[7:0]
VGA_CLK
VGA_BLANK_n
VGA_SYNC_n

Access Parameters:

- Modeled/Inspired by TMS9918 and NES PPU
- 32bit data, 16bit address
- “Virtual” Address Space

Valid Address Ranges	Table
0x0000 -> 0x03F	Attribute Table
0x1000 -> 0x102F	Color Table
0x2000 -> 0x24BA	Pattern Table
0x3000 -> 0x3FFF	Sprite Table

```
logic [3:0] mem_write;
assign a_addr = mem_write[0] ? w_addr[5:0]: ar_addr[5:0];
assign c_addr = mem_write[1] ? w_addr[5:0]: cr_addr;
assign p_addr = mem_write[2] ? w_addr[11:0]: (state == OUTPUT ? ( (vcount[9:4] *
40) + hcount[10:5]) : vcount[9:4] * 40);
assign s_addr = mem_write[3] ? w_addr[11:0]: (state == OUTPUT ? ( (pattern[7:0] <<
4) + vcount[3:0]): sr_addr);

memory #(32,  64, 6)    attr_table  (.clk(clk),  .we(mem_write[0]),
.addr(a_addr[5:0]), .data_in(w_data), .data_out(sprite_attr));
memory #(32,  64, 6)    color_table (.clk(clk),  .we(mem_write[1]),
.addr(c_addr[5:0]), .data_in(w_data), .data_out(color_out));
memory #(32, 2048, 11)  pattern_table(.clk(clk), .we(mem_write[2]),
.addr(p_addr[10:0]), .data_in(w_data), .data_out(pattern));
memory #(32, 4096, 12)  sprite_table(.clk(clk),  .we(mem_write[3]),
.addr(s_addr[11:0]), .data_in(w_data), .data_out(sprite));

always_ff @(posedge clk) begin
        mem_write <= 3'b0;
        if (chipselect && write) begin
                case(address[15:12])
                        4'b0000: mem_write[0]   <= 1'b1;
                        4'b0001: mem_write[1]   <= 1'b1;
                        4'b0010: mem_write[2]   <= 1'b1;
                        default: mem_write[3]   <= 1'b1;
                endcase
                w_addr <= address;
                w_data <= writedata;
        end
end
```

Graphics Display - Memory Budget

Each pixel location is 2 bits wide (to index into the color table!)

Category	Image	Size (pixels)	Variants	Total Bits
Duck		32 x 32	4 (up, down, dead, flying away)	4 * 32 * 32 * 2 = 8,192 bits
Bullet		16 x 16	1	1 * 16 * 16 * 2 = 512 bits
Background		16 x 16	8	8* 16 * 16 * 2 = 4096 bits
Score + Round (numbers)		16 x 16	10	10 * 16 * 16 * 2 = 10240 bits
Crosshair	X	16 x 16	1	1 * 16 * 16 * 2 =
Color Pallet		4 x 32 (bits not pixels)	8	4 * 32 = 128 bits

Total Budget Required: 23,680 bits = 2,960 bytes = 2.89 KB

Graphic Generation

- Automatically Generated Sprite Data Via Python Scripts!

Core Parameters

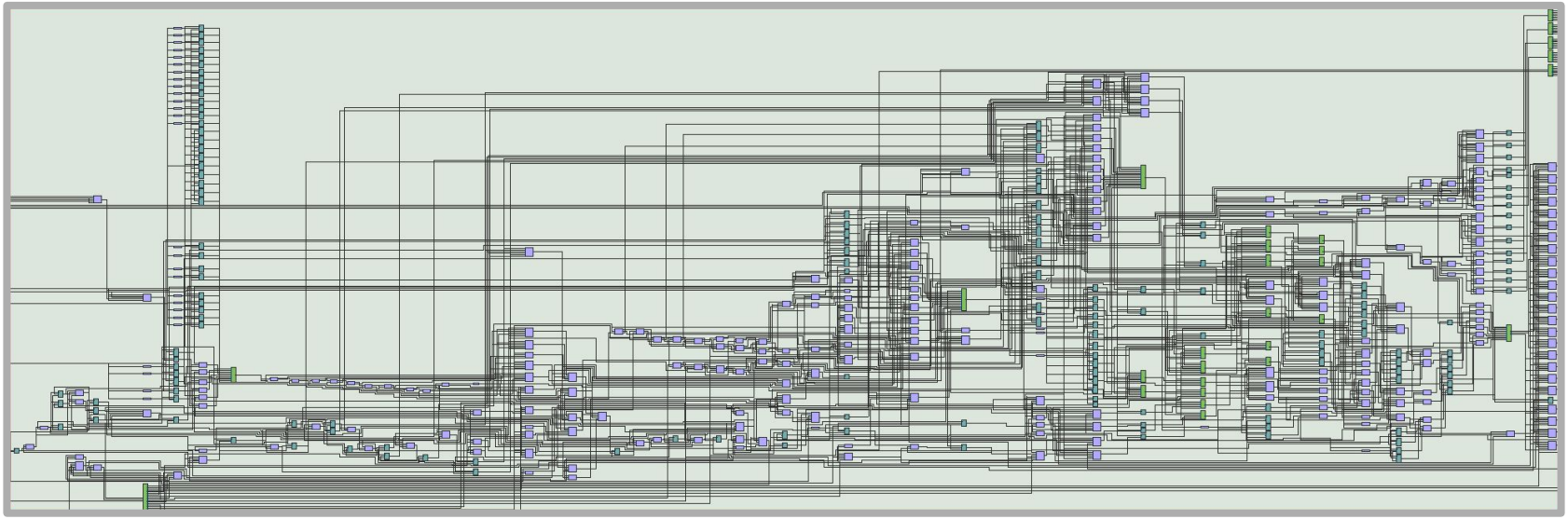
- x, y location of each sprite (20 bits)
- Sprite table for each sprite (8 bits)
- Color table for each sprite (4 bits)

Param	Y Location	X Location	Sprite Table	Color Table
Bit Location	0 - 9	10-19	20-27	28-31

Core Parameters

- Duck 1
- Duck 2
- Score Digit 1
- Score Digit 2
- Round
- Bullet 1
- Bullet 2
- Bullet 3
- Crosshair

Cool Diagram of PPU's datapath



Software

- Sprites.c
  - Used to load attributes, patterns, color palettes, and sprites into PPU at startup and during runtime
- Game.c
  - Handles game mechanics such as gamestate, scoring, and duck movement.
- Duck_hunt.c
  - Main game loop.
  - Setups wiiuse library and handlings user input

Game Logic

- Skeet Shooting Style Game with Multiple Rounds
- Bird Speed Increased as Game Progresses
- Birds Spawn From Fixed Location in Random Directions
- Player Has 3 Bullets to Shoot 2 Ducks Per Round
- If Player Shoots All Bullets or Misses All Ducks, The Next Round Begins
- After 8 Rounds The Game is Over

Inputs

- Wii Remote connected via Bluetooth
- WiiUse Library
  - Handles connecting to Wii Remote and handling inputs
  - duck_hunt.c uses wiiuse library to poll wii remote
- Custom Kernel built to support bluetooth drivers
- Press A to Start Game
- Move Crosshair and shoot

Future Work

- Better Start Game and End Game Screens
- More Varied Difficulty Levels Selectable by Players
- More Sprites!
  - Duck Shocked
  - Duck Mid Flap
  - A dog!

Thank you!