

CSE 4117

Final Project

In this project, you will use Bird-CPU to read keystrokes from a ps2 keyboard and draw a scene on a VGA monitor.

Your system will start by drawing a red square on a green background. The red square will be 11 pixels wide and 11 pixels high, and will be exactly at the center of the screen.

--When the key W is pressed, the square will move 4 pixels up.

--When the key S is pressed, the square will move 4 pixels down.

--When the key A is pressed, the square will move 4 pixels left.

--When the key D is pressed, the square will move 4 pixels right.

When the keys are released, no extra movement will occur.

Use the ps2 keyboard, Bird CPU (with 12-bit address bus), VGA monitor, and Bird assembler codes given in the lecture notes.

Modify them suitably, and fill in the blank parts.

On the hardware side, you have to write the top module which "glues" the CPU, keyboard, and monitor. Assign addresses to the keyboard and monitor registers.

You are also required to modify the VGA code slightly. You can use Bird-CPU and keyboard codes without modification.

On the software side, you have to write the assembly code which manages the hardware for the given task. Use the bird assembler given in the lecture notes to assemble your software.

HINT:

=====

In the code segment given below, the bird CPU reads from the keyboard and prints to a 7-segment display. You can get some ideas on how to write your top module from this code segment.

```
//Main module
```

```
module main_module ( input clk, input ps2c, input ps2d, output logic [3:0] grounds, output logic [6:0] display, input pushbutton //may be used as clock );
```

```
logic [15:0] data_all;
```

```
logic [3:0] keyout;
```

```
logic ack;
```

```

//memory map is defined here
localparam BEGINMEM=12'h000,
           ENDMEM=12'h1ff,
           KEYBOARD=12'h900,
           SEVENSEG=12'hb00;

// memory chip
logic [15:0] memory [0:127];

// cpu's input-output pins
logic [15:0] data_out;
logic [15:0] data_in;
logic [11:0] address;
logic memwt;

sevenssegment ss1 (//to be added);
keyboard kb1(//to be added);
bird br1 (//to be added);

//multiplexer for cpu input
always_comb
    if ( (BEGINMEM<=address) && (address<=ENDMEM) )
        begin
            data_in=memory[address];
        end
    else if (address==KEYBOARD+1)
        begin
            //to be added
        end
    else if (address==KEYBOARD)
        begin
            //to be added
        end

```

```

        end
    else begin
        data_in=16'hf345; //any number
    end
//multiplexer for cpu output
    always_ff @(posedge clk) //data output port of the cpu
    if (memwt)
        if ( (BEGINMEM<=address) && (address<=ENDMEM) )
            memory[address]<=data_out;
        else if ( SEVENSEG==address)
            data_all<=data_out;

initial
begin
    $readmemh("ram.dat", memory);
end
endmodule

```

Submission Instructions

- You should submit your homework as a zip file which contains all source files.
- The name of your zip file will be as
CSE4117_name_surname_ID1_name_surname_ID2_name_surname_ID3_hw#.zip.
- It is enough, if a single group member uploads.
- Late submissions will loose 10 points for each day after the deadline.