CSC258 Project Proposal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | First Name | Last Name | Student No. | Email |
| Member 1 | Tiago | Ferreira | 1004305245 | t.ferreira@mail.utoronto.ca |
| Member 2 | Pan | Chen | 1005430776 | pan.chen@mail.utoronto.ca |

What is the title of your project?

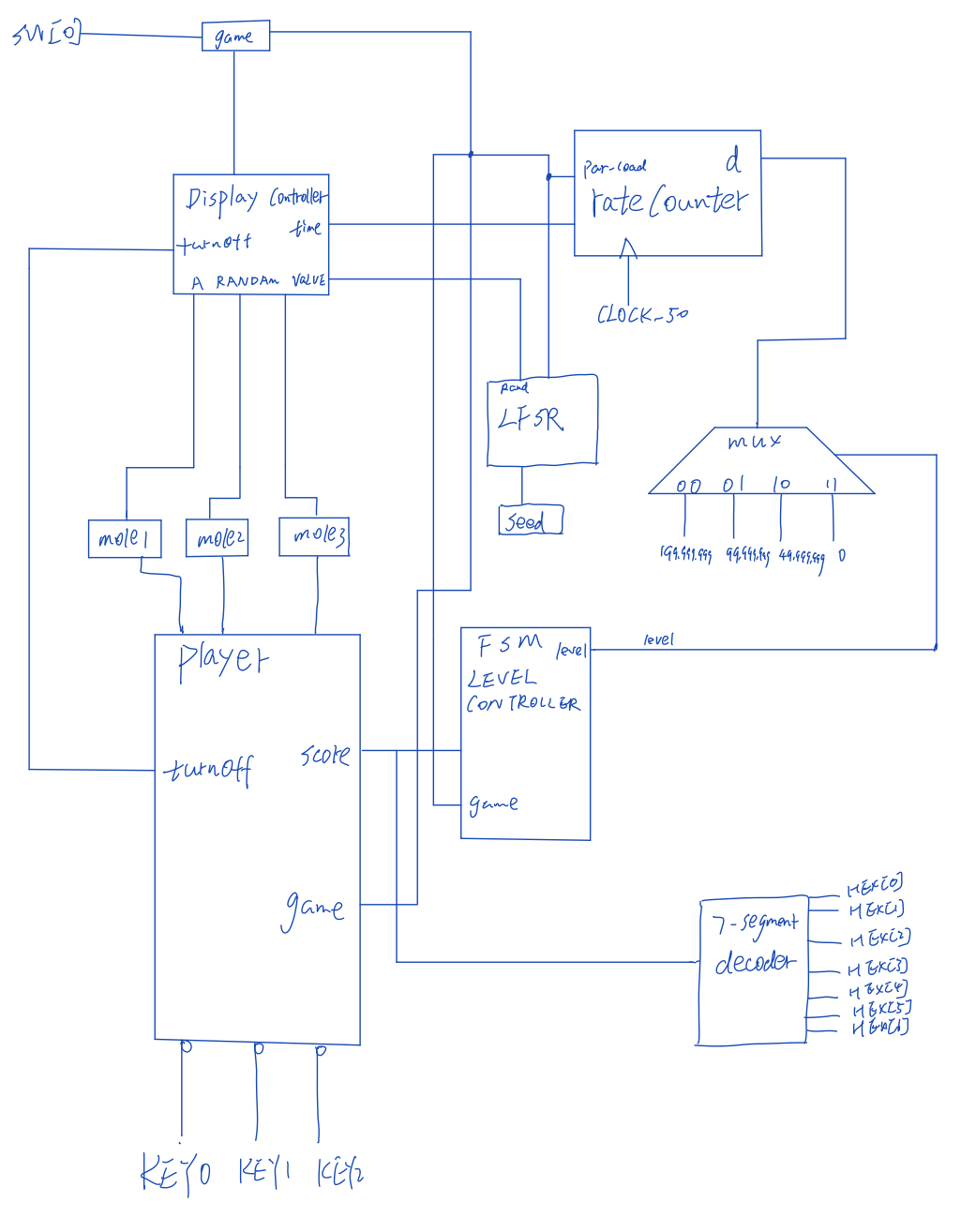
**Whack a Mole**

Provide a one-paragraph description of your project.

This project is a game. The game will fire LED’s at random times each level and the user must click the corresponding button to “whack the LED”. The game will feature a fixed speed and a fixed number of moles. The goal is to whack as many LED’s as you can. If you take to long, then you will not receive the point. Your score is then displayed on a HEX-display and represents the amount of LED’s you “whacked”.

Project Description:

**Design Schematic (For the core part (except VGA part)):**



**Design Specifications:**

1. A fixed number of LEDs (moles) will be on for a fixed length of time for each state. If the player pushes the corresponding key before the LED automatically turns off, the player earns a certain number of points.
2. rateCounter from lab5 will be used to control when the LEDs should turn off.
3. When player clicks the “start” signal, the game will start.
4. After the player earns a certain number of points, he will enter the next state, with higher speed. (There are four different states.)
5. **Player will use A S D in the keyboard as their ‘Whacker’.**
6. **The game (the moles, the score …) will be displayed on the screen of the lab computer by VGA.**

**Pseudo-code of algorithm**

For each LED’s turnning on / off controller

If start

Choose a random LED to display

While (rateCounter != a\_fixed\_frequency)

LED keeps on

LED turns off.

**For player gaining scores controller**

If LED[0] turns on and KEY[0]:

Score += this\_state\_point

turnoff = 1

If LED[1] turns on and KEY[1]:

Score += this\_state\_point

turnoff = 1

If LED[2] turns on and KEY[2]:

Score += this\_state\_point

turnoff = 1

If LED[3] turns on and KEY[3]:

Score += this\_state\_point

turnoff = 1

**Frequency change controller (like a FSM):**

if score == level1Max or level2Max or level3Max:

update = 1

else

update = 0

State 1

If update:

Go to State 2

State 2

If update:

Go to State 3

State 3

If update:

Go to State 4

What will you accomplish for the first milestone?

We will first implement the module (Display Controller) that chooses a random LED to display. We will use the Linear-feedback shift register to do this part of work.

Also, for the first milestone, we will add feature to this module which turns off the LED after a certain time even if the player doesn’t push the corresponding key. (In this case, the player doesn’t earn any point). We will use the rateCounter from lab5 to do it.

And we will make the “mole” turns off if the player clicks the right button and also we will let the player earn 1 point if they do so.

And the players will lose 1 point if they click the wrong button. The player who has 0 point will be exempted.

And for the first milestone before FSM is implemented, we will set the display time 2 seconds and wait time between two tries 3 seconds.

We will show the simulation waveform as part of this milestone and we will try to make both display module and player module work on the FPGA board.

What will you accomplish for the second milestone?

We will implement out vga module so that it draws three separate blocks/moles and so that it can make them disappear when needed. We will show the TA the code for this module as part of the second milestone and **we will attempt** to show it working on the VGA adapter using switches as inputs.

We will also write the module for keyboard input and we will again show the TA our code for it as part of the second milestone. **We will attempt** to show its working using the PS2 board and LEDS, using A S D as input from the keyboard.

What will you accomplish for the third milestone?

* For this milestone we will try to fix any issues with the VGA that we had in the last milestone. If there were no errors, then we will try to change the blocks on the vga into a sprite.
* We will also implement the FSM that adjusts the game difficulty (the mole display time) according to how many points the player has earned.
* We will also fix an issues we had from previous milestones
* Finally we will connect all of our modules together into the final game and show the working project to the TA!

How does this project relate to material covered in CSC258?

This project will use multiple topics covered in CSC258. It will require sequential circuits to design. It will represent a FSM with multiple states. It will feature a HEX display for score output. It will use clock frequency for random LED Generation and other things. It will use other modules and topics learned in this class. **And it will use VGA knowledge to put our game on the screen and use a keyboard to control game.** Finally, it will be programmed in Verilog which is a big part of the course.

What's cool about this project (to CSC258 students and non-CSC258 students)?

To CSC258 students it is cool to them because it is a unique idea that we have not seen in past 258 projects. It will require unique knowledge of what was learned in this class, such as rateCounter, VGA, how to use a keyboard to control signals …. To non-CSC258 students it is cool to them because they will not understand the process to actually make the game so they will be amazed and wonder how it is possible to actually make something like this using hardware.

Why does the idea of working on this appeal to you personally?

We used to have whack-a-mole when we were younger kid and this project gets its idea from that. It is interesting to us to see how we can algorithmically implement this using circuit and hardware knowledge.