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?

(e.g. Laser-Triggered Music Box)

Whack-a-LED

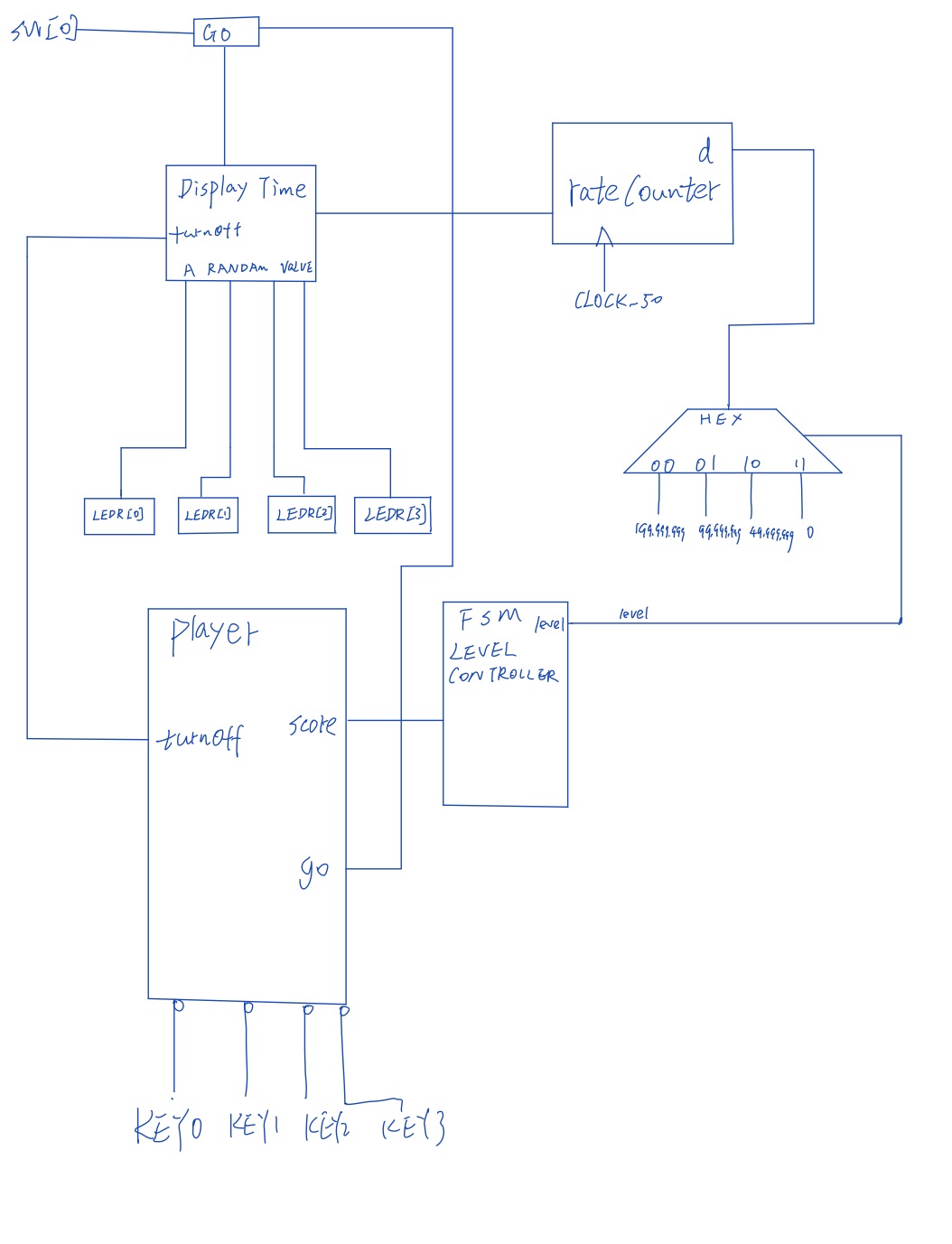
Provide a one-paragraph description of your project.

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:

Whack-a-LED

Design Schematic:



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 switches (or key?) before the LEDs turn 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 and / or more LEDs being on at the same time. (There are four different states.)

STEP1 : Algorithmic modelling

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

rateCounter0 = Max (in order to turn off the LED)

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

Score += this\_state\_point

rateCounter1 = Max

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

Score += this\_state\_point

RateCounter2 = Max

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

Score += this\_state\_point

RateCounter3 = Max

Frequency change controller (like a FSM):

State 1

If score == XXX

* State 2

State 2

If score == XXX

* State 3

State 3

If score == XXX

* State 4

(This is where you describe your project in detail. You can use the Design Case Studies slides as a reference on how to create the following components for your project. All of these components are not compulsory for you to have but most projects usually have these components. They are: high level pseudo code, state diagram, datapath-and-control-CU block diagram, input/output block diagram [Example: <https://www.nandland.com/goboard/images/project10-pong-block-diagram.png>], truth tables etc. These components should be designed and described to show your understanding of your design i.e. how many bits is each input/output, what is the max number your counter can count up to, how many counters/shift registers you need to use etc. A good idea is to get an initial draft of this done and show to your TA in the next lab or during office hours to get feedback.)

What will you accomplish for the first milestone?

We will first implement the module that chooses a random LED to display. Also in this module we will use the rateCounter from lab5 so that the LED can be off even if the player doesn’t push the corresponding key. This is the most crucial part of our project.

(Advice here: Pretend that you're designing Lab 8 around your project idea, in keeping with the difficulty level of the previous labs. Try to be **specific and detailed** in describing the components that you will complete. Don't say that you'll "think about" or "plan" or "design" something.

Bad example 1: We build the graphical interface.

Good example 1: We display moving spaceships on VGA.

Bad example 2: We write code for the PS2 keyboard interface.

Good example 2: We make the PS2 keyboard work and show the key inputs on the HEX display.

Assume your project can be developed in three independent parts, what you write in the space below should outline the components of the first part. Make sure to describe a full lab's worth of work, including the evidence of your work that you will provide to the TAs to justify getting the full marks for this milestone.)

What will you accomplish for the second milestone?

(similar advice as above, but for the second part of your project. Remember to specify what inputs and outputs will be used for each milestone. If your project is a visual game for example, what will appear on the screen for each milestone, for example, static colored boxes in one milestone and moving boxes in the next one etc.)

We will implement the module that can turn off the LED when the player pushes the right key within the limited time. Also, this module will give player some points for their ‘whacking’ the LED in time.

And depending on our speed of implementing this part, we may implement the module for the finite state machine for this milestone as well. That machine controls the level the player is on. For example, if the input to the finite machine reaches a certain value, the game will move to the next level, with shorter display time. More challenging to the player.

What will you accomplish for the third milestone?

(don't say "everything" just because this is the final milestone; describe the final components instead, and exactly what the TAs should expect to see)

We will implement the module for the finite state machine for this third milestone. What’s more, we will debug our codes. And if everything goes smoothly, we might add more challenging levels to the player to make our game more challenge and more interesting.

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. 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. 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.