Cassey Shao
Valentina Manferrari

GINGER vs. MOUSE: Operating Instructions

1. Game Purpose and Brief Description

This maze game is based on the storyline of Tom and Jerry. In our game, the protagonist is the mouse (i.e. the player) and it must escape from the labyrinth within the time constraint, and without getting caught by the antagonist cat named Ginger (i.e the computer).
The player must use the switches to navigate the maze, and they win when the mouse safely escapes and eats the whole block of cheese on the other side of the maze.

2. DE1-SoC input/output ports used

- ➢ [2:0] KEY used to start and reset levels.
- ➢ [3:0] SW used to move the mouse up/down/left/right.
- ➢ [1:0] LEDR used to display the current level.
- ➢ [3:0] HEX used to countdown time.
- ➢ The VGA used to display the game.

3. Warnings on CPUlator that need to be disabled

- ❑ Device-specific warnings
- ❑ Function nesting too deep
- ❑ Interrupt nesting too deep

4. Instructions to play

1. Read the instructions shown on the start display.
2. Press KEY[0] to start playing *Level 1* when you are ready.
3. A maze with the cat Ginger in it will appear; the mouse and the cheese are on opposite sides. Use SW[3:0] to help the mouse escape from the maze and eat the cheese.
   Turn on SW[0] to go left, SW[1] to go right, SW[2] to go up and SW[3] to go down.
   a. Make sure to avoid the hungry cat Ginger, who is moving through the maze. If you meet him, you are done for! You will need to redo the level.
   b. You have 2 minutes and 30 seconds before the poor mouse starves to death, hurry! up! The hex display shows you how much time is left (MM:SS).
4. If you passed *Level 1,* a congratulations panel will appear.
   → Press KEY[1] to go to *Level 2.* (Jump to step 5)
   If you got caught by the cat Ginger or you ran out of time, a game over panel will appear.
   → Press KEY[2] to play again *Level 1*. (Go back to step 3)
5. Very good, now you are on *Level 2*. A new maze will appear, the cat Ginger is still in it, and is still hoping to catch the mouse this time!

It works the same as *Level 1*, but this time you need to be faster! The mouse only has 1 minute and 30 seconds to reach the cheese without encountering Ginger. The countdown is shown on the HEX Display (MM:SS).

6. If you get caught by the cat, or run out of time, a game over panel will appear.
   → Press KEY[2] to play *Level 2* again. (Go back to step 5)
7. You win, congratulations! "The End" display is shown; recompile the program to play again.

Note: If you forget which level you are currently playing, you can check the LEDs. If LEDR[0] is on, it means you are playing Level 1. If LEDR[1] is on, it means you are playing Level 2.

References

[1] ECE243 Winter 2020 - Lab 7

[2] W. D. Pullen, *Think Labyrinth: Maze Algorithms*. [Online]. Available: http://www.astrolog.org/labyrnth/algrithm.htm. [Accessed: 09-Apr-2020].

[3] TheJollySin, "What's a good algorithm to generate a maze?," *Stack Overflow*. [Online]. Available:https://stackoverflow.com/questions/38502/whats-a-good-algorithm-to-generate-a-maze. [Accessed: 09-Apr-2020].

[4] A Lone Coder, "Programming Mazes", *YouTube*. [Online]. Available:https://www.youtube.com/watch?v=Y37-gB83HKE [Accessed: 09-Apr-2020].

[5] G. Cope, *Maze Generation Algorithm - Depth First Search*. [Online]. Available: https://www.algosome.com/articles/maze-generation-depth-first.html. [Accessed: 09-Apr-2020].

Attribution Table

The division of the amount of work contributed by each team member is 50%-50%.

Name	**Contributions**
Cassey Shao	● Created and plotted drawings on the VGA display (start screen, game screen that has the cheese, cat, mouse, game over screen, cat catches mouse screen, time is up screen) ● Connected the LEDs, KEYs, and SWs to control game logic ● Animated the mouse and cat ○ Moving within the walls of the maze logic ● Implemented logic between start screen to level one screen to level two screen to game over
Valentina Manferrari	● Created and plotted the maze on the VGA display (implemented an algorithm to generate random maze so that the game is different each time you play it) ● Implemented a timer on the HEX display ● Coordinated the timer to run parallel with the game ● Implemented the game logic for cat catches mouse and time is up screen