Homework 4: Snake
October 25, 2017
Sabbir Ahmed

1 Background

For this assignment, a version of the classical snake game will be implemented.

The game was to conform to the following specifications:

- The game play area is set up as a 26-by-38 grid surrounded by an electric fence (blue). The game display should occupy the majority of the screen.
- To start the game, press RESET/BTN_SOUTH. Upon release, a single grid point represents a snake (green) and a single grid point represents food (red).
- The snake starts by moving to the right one grid point roughly every 1/8th of a second (125 ms update interval).
- A player may change the direction of the movement by 90 degree clockwise or counter-clockwise by using the rotary switch on the FPGA board. Every position (notch) change of the rotary switch should correspond to a 90 degree angle change.
- If the snake goes onto the fence, the game should freeze.
- If the snake goes onto the food, the length of the snake should increase by one grid segment on the next movement with a trailing body, per the behavior of the game shown in the provided link. The moving head be green, and the grown body should be cyan (green+blue).
- Each time the snake reaches the food, another food should be positioned in a manor seemly random to the user.
- Each subsequent time the snake eats food, the segment length should grow by one, up to a length of 32 (including the head).
- If the head of the snake overlaps a body segment then the game should freeze.
- If the length of the snake reaches 32, the game should instead increase in speed by reducing the update interval by roughly 10 ms.

2 Design Approach

Several discrete modules will be used to implement the game: `igniter` , `extinguisher` and `candle_controller` . These submodules were connected and debounced using a top level module that may be visualized with the schematic diagram configured as a block diagram in Figure 1.

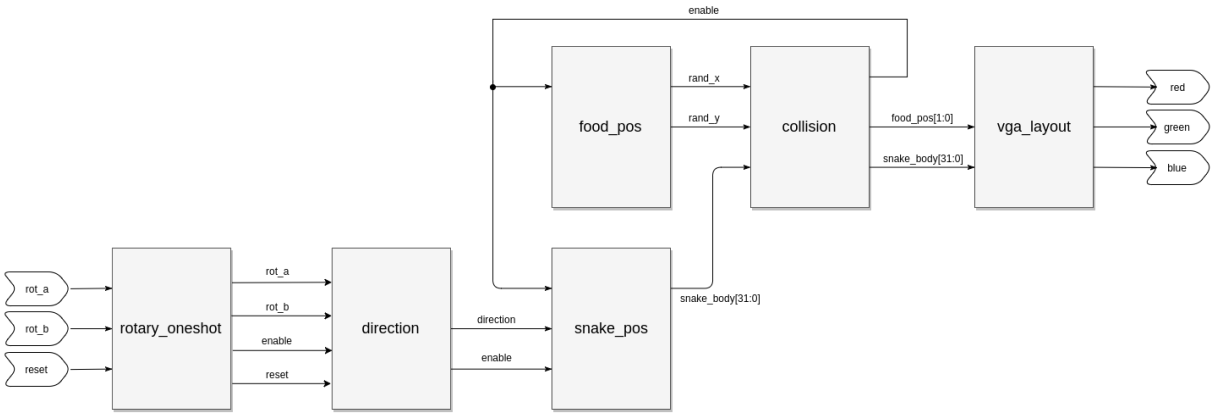

Figure 1: Schematic of the Implementation of the Game