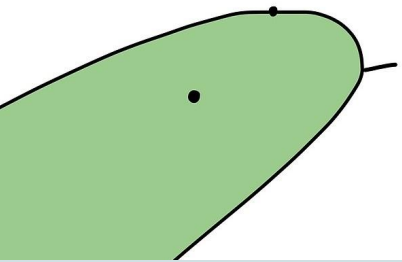
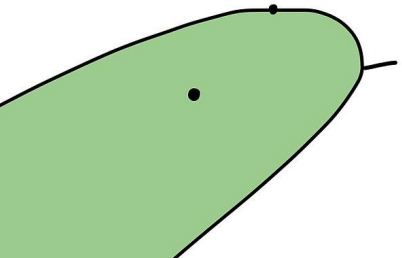
SNAKE



Aaron & Joani

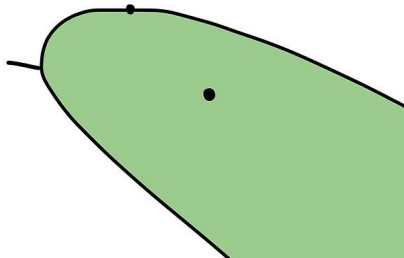
Gameplay

- Ordinary Implementation of the Classic Game Snake
- Win when the length of snake is 100
- Lose if the head of the snake hits any of the borders or itself



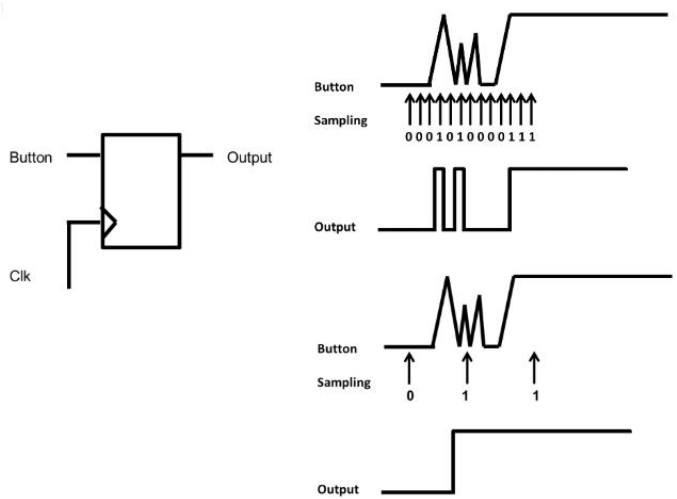
Verilog Overview

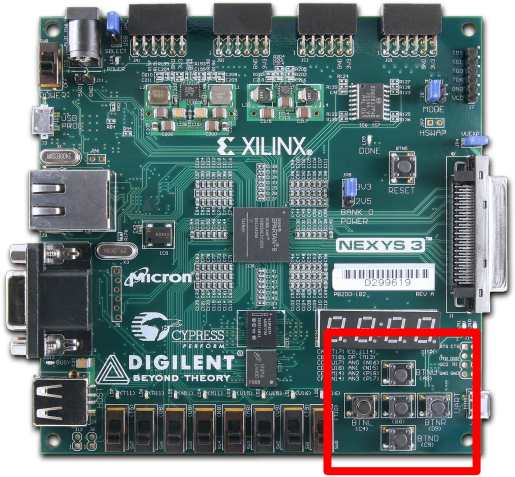
- top_snake.v
- debouncer.v
- snake.v
- clk_div.v
- update_clk.v
- random_food.v
- vga640x480.v
- score_display.v
- seg_display.v



Debouncer

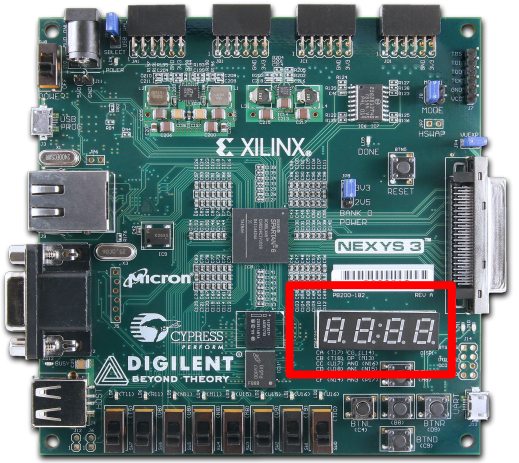
- Used to debounce the reset and direction buttons
- Used the same technique as previous labs

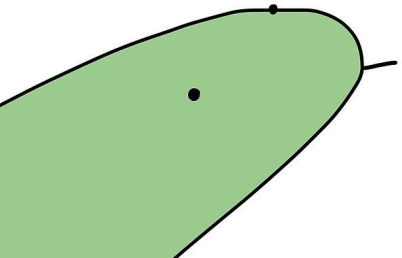




Clock Divider

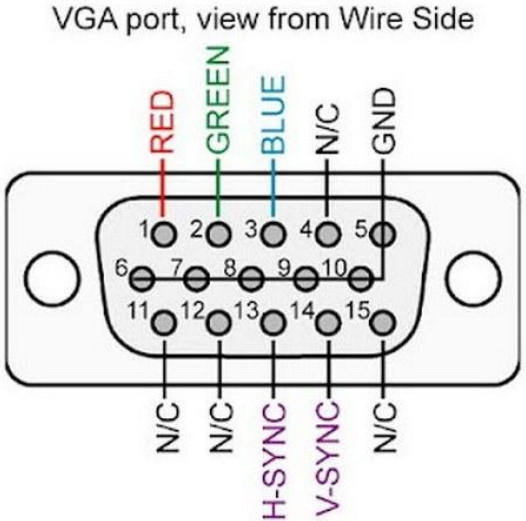
- **Fast Clock** – 500 Hz
  - Used for seven segment display
- **Blink Clock** – 0.5 Hz
  - Used for blinking the seven segment display

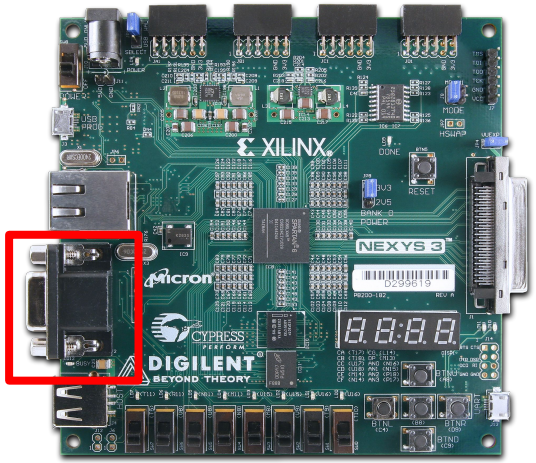


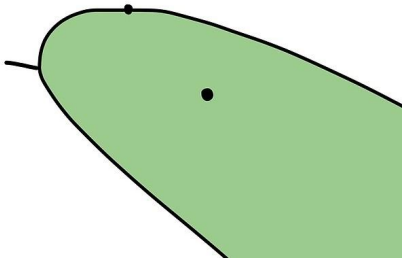


VGA Display

- Used to display the game on the computer screen
- Source: https://timetoexplore.net/blog/arty-fpga-vga-verilog-01





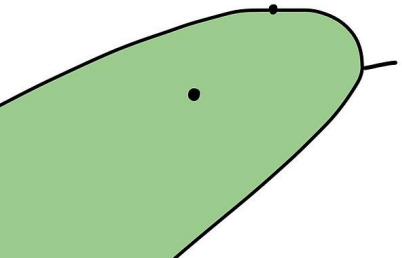


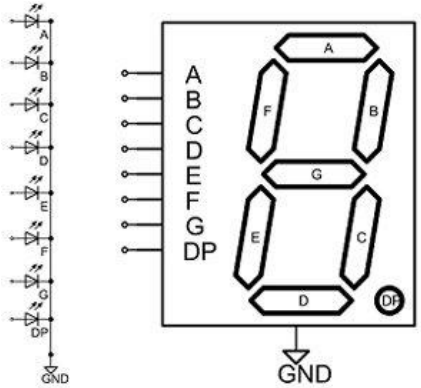
Score Display

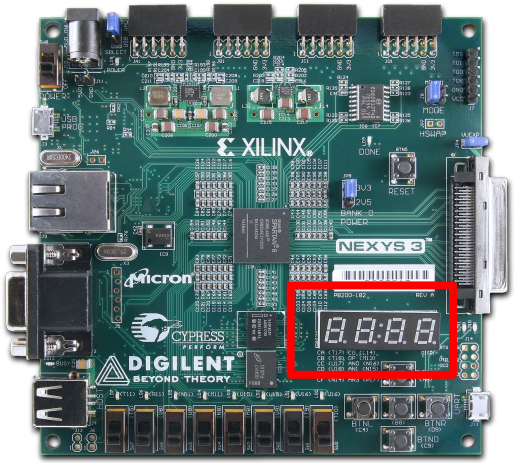
- Takes in the 8-bit size register from the game implementation and converts it into three 4-bit registers representing numbers from a range 0-9, so that the score can be displayed on the seven segment display
- Since the max. length of the snake is 100 blocks, the first digit in the seven segment display will be zero

Segment Display

- Took in a desired output, which were either numbers or several letters, and assigned either a one or a zero to each segment in order to display the desired output
- Each segment is active-low

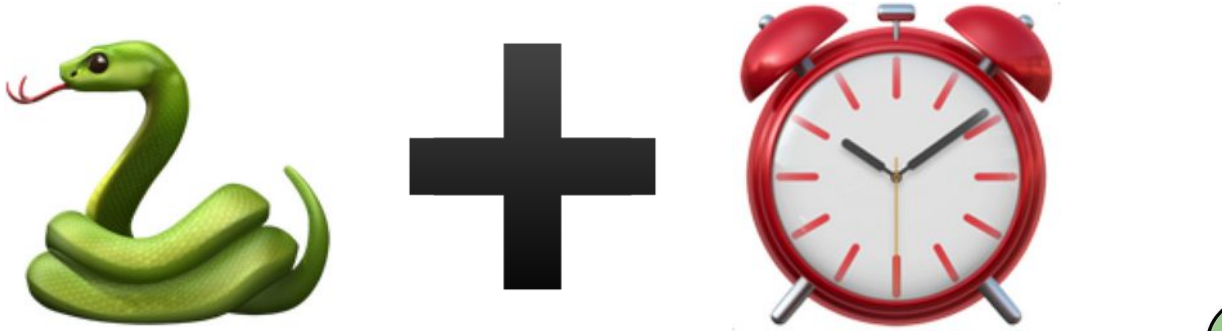






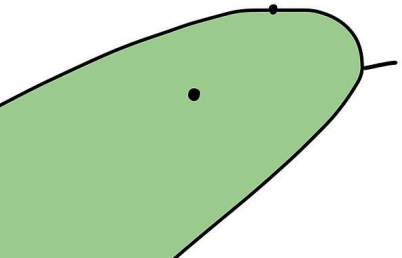
Update Clock

- Creates a strobe at 8 Hz, which is responsible for moving the snake consistently by 10 blocks



Random Food Generator

- Generated two random coordinates
- Pseudo-random as it used a mod counter to come up the coordinates
- Ran at 100 MHz



Snake

- Responsible for the gameplay implementation
- Modules included in snake.v are:
  - update_clk.v
  - random_food.v
  - vga640x480.v
  - score_display.v
  - seg_display.v

LIVE DEMO