

Snakes and Obstacles

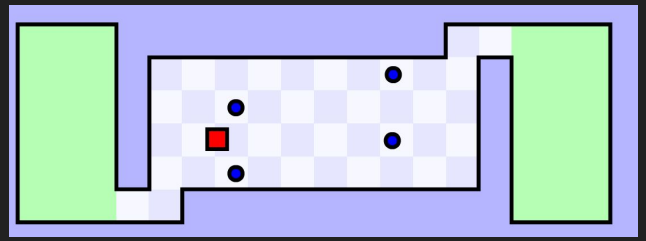
CSC258: Final Project


Vikram and Hsiang-chi
Wednesday Lab Section
Station #87

The Proposal Overview (Post Feedback)

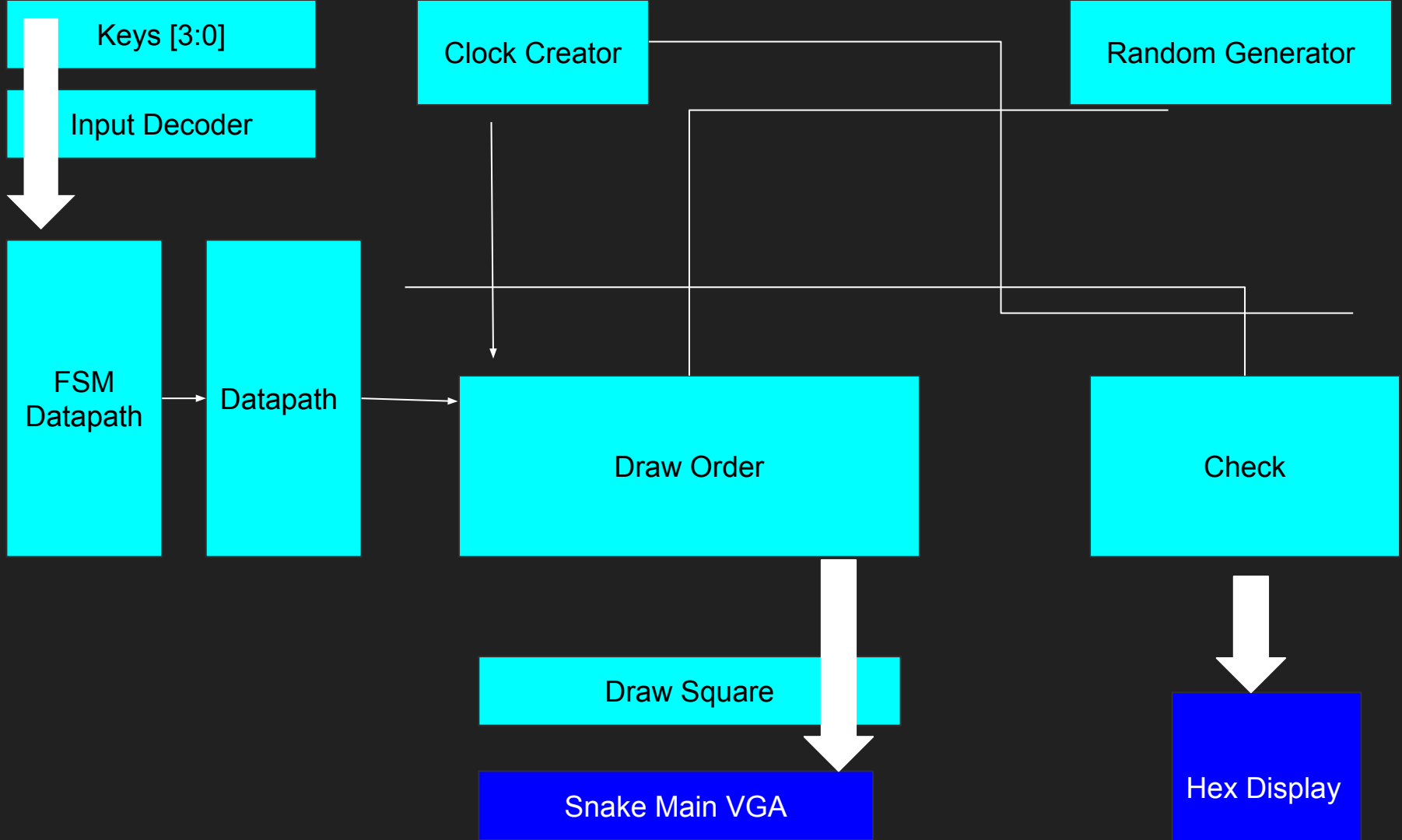
A twist on the classic game snakes meshed with the world's hardest game

- Snake moves around the board collecting food and tries its best to avoid barriers.
- A counter keeps track of the score and increases the difficulty of the game as points are gained.
- During Feedback: Told that given the time frame it is sufficient to work with a fixed size for the snake and increase the difficulty of the game with the obstacles.

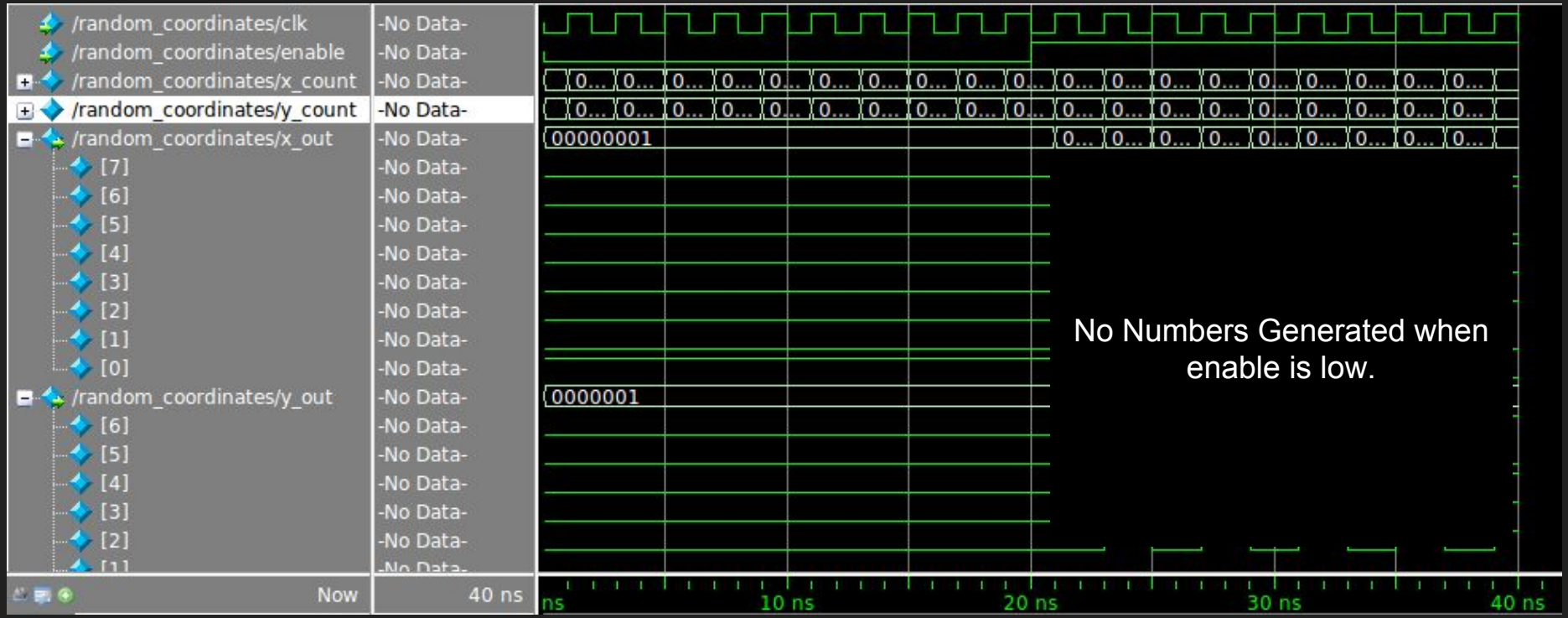


Overview of Design

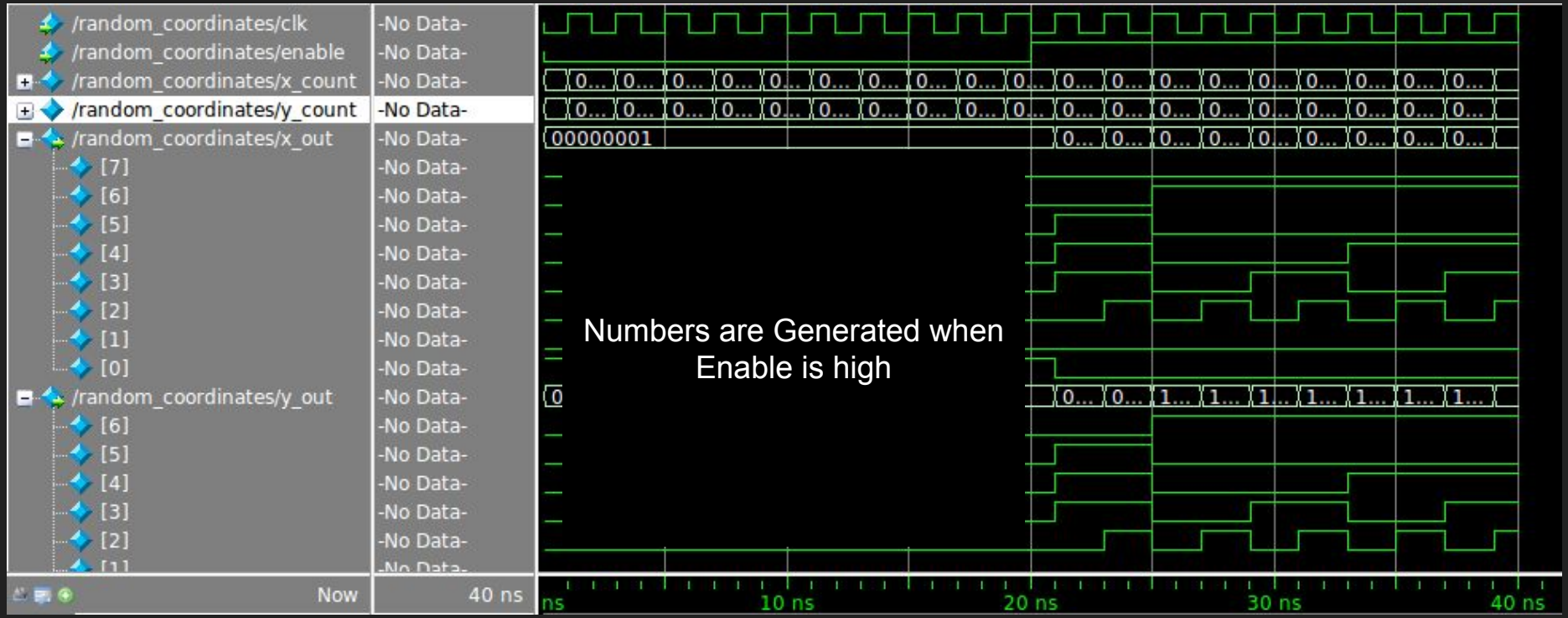
1. SnakeMain takes in coordinates and colour and displays to the VGA
2. Hex Decoder preps 4 digit binary number for Hex Display.
3. Control/FSM signals that drive the Datapath
4. Datapath drives the movement of the Snake
5. Clock Creator used to create new clocks
6. Input Decoder Uses latches to store user input and converts them to signals that drive the FSM
7. Check deals with scenarios involving snake hitting something.
8. Random Coordinates generates coordinates for food.
9. Bar Used to construct a new O=obstacle
10. Draw Order cycles through the multiple objects and draws them one by one.
11. Draw Square takes a point and constructs a 4x4 square.


Keys [3:0]
Input Decoder
Clock Creator
Random Generator
FSM Datapath
Datapath
Draw Order
Check
Draw Square
Snake Main VGA
Hex Display

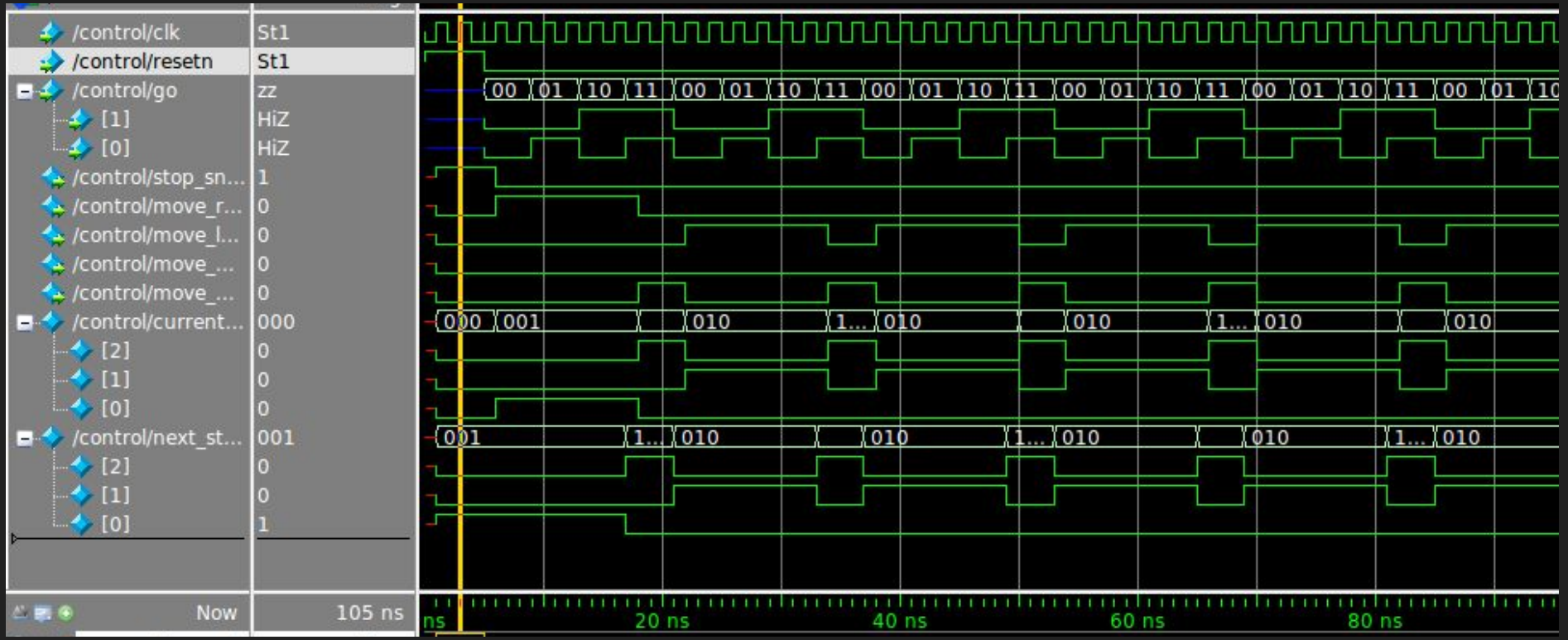
Coordinates



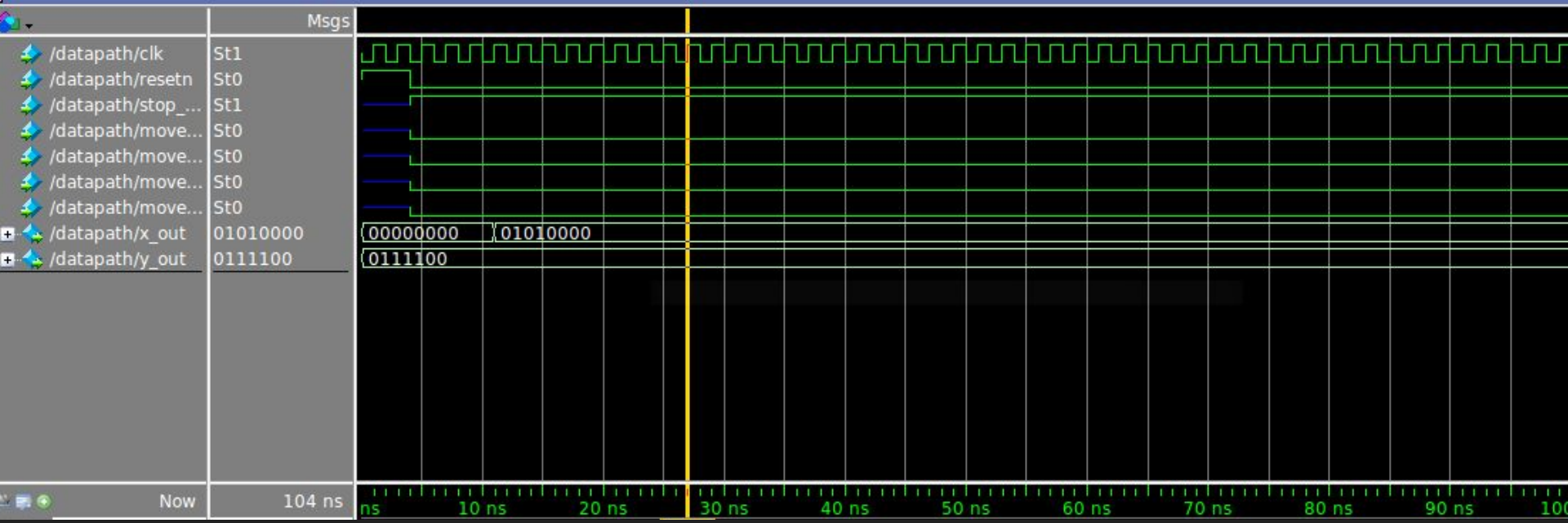
Coordinates



Control / FSA


/control/clk
St1
/control/resetn
St1
/control/go
zz
[1]
HiZ
[0]
HiZ
/control/stop_sn...
1
/control/move_r...
0
/control/move_l...
0
/control/move_...
0
/control/move_...
0
/control/current...
000
[2]
0
[1]
0
[0]
0
/control/next_st...
001
[2]
0
[1]
0
[0]
1
Now
105 ns
ns
20 ns
40 ns
60 ns
80 ns

Datapath


Msgs
/datapath/clk
/datapath/resetn
/datapath/stop_...
/datapath/move...
/datapath/move...
/datapath/move...
/datapath/move...
/datapath/x_out
/datapath/y_out
St1
St0
St1
St0
St0
St0
St0
01010000
0111100
00000000
01010000
0111100
Now
104 ns
ns
10 ns
20 ns
30 ns
40 ns
50 ns
60 ns
70 ns
80 ns
90 ns

Work Breakdown

	Snake Design	FSM Design	Removing Trace	Input Decoder	Datapath Design	
Week 1	Vikram	Vikram	Jackie	Vikram	Jackie	

	Snake touch itself	Creating Clocks	Scorekeeping HEX Display	Constructing Obstacle	Food Random Location	Snake on Obstacles
Week 2	Jackie	Jackie	Vikram	Vikram	Vikram	Jackie

Week 3: Was spent Fixing bugs and implementing levels (together).

Major Challenges Faced

- One pixel from the previous clock cycle would keep lagging behind.
- Modules were set to the wrong clock.
- Bar would leave a box behind every 10 pixels
- Keyboard Input was not getting captured - forgot to construct a latch.

What we Learned

In General, Clocks can be confusing.

- Creating Multiple Clocks with one clock makes each successive clock less accurate.
- Sometimes for synchronicity sake it is better to pass values through a module even if they are unchanged. Specifically when there are multiple clocks involved.

Order Matters

- When designing a game of snake, drawing a black square first and then a coloured square is different from drawing the coloured square first.

What we Learned

LEDR can be a useful debugging tool

- Lighting up the LEDs with the signals helps trace when something is going wrong