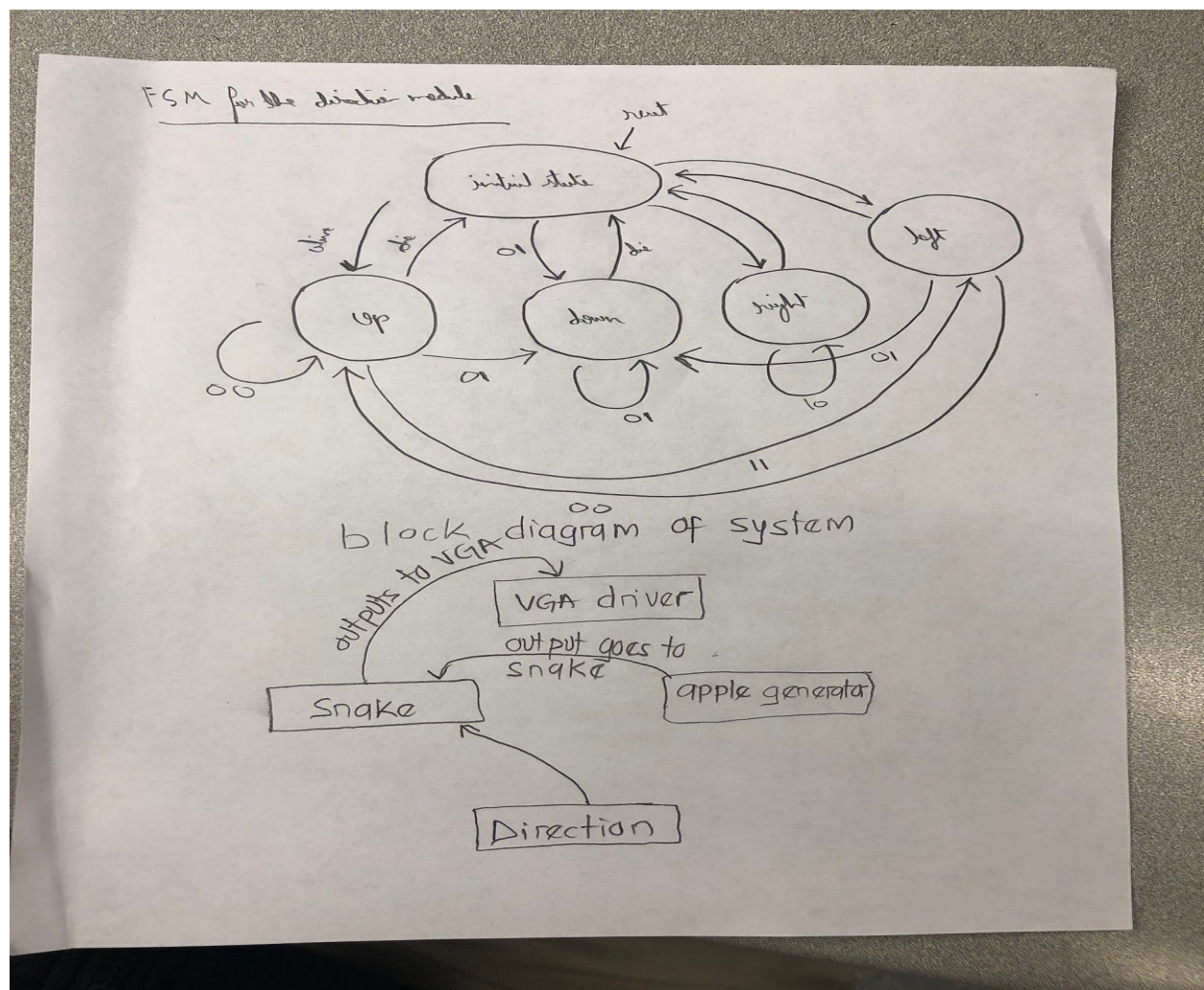


Abdiasis Ibrahim & Umair Liaqat

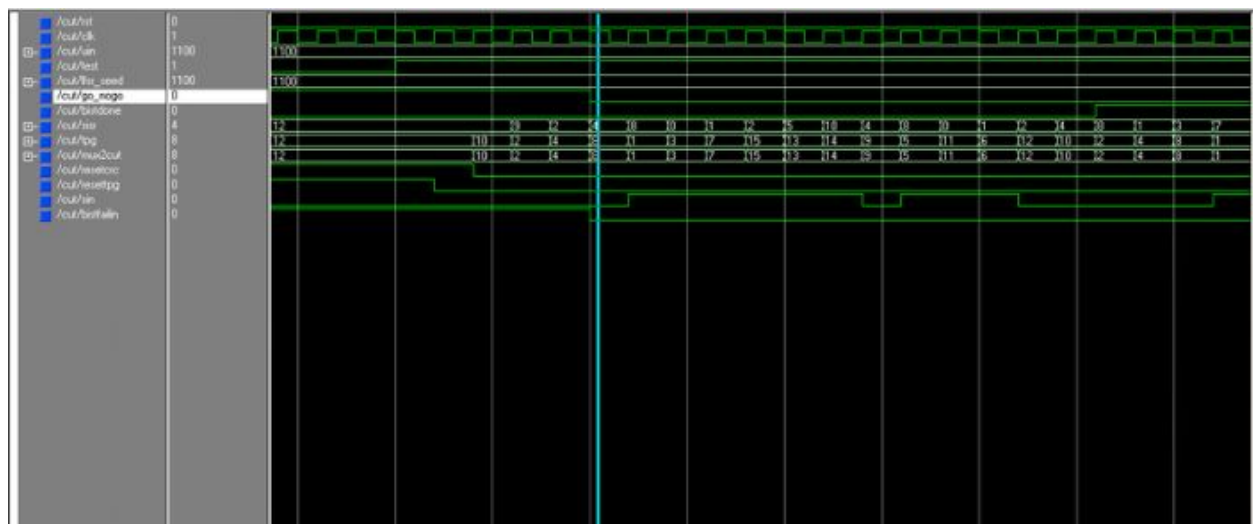
Procedure:

We divided the lab into three modules: the apple module which was generated randomly using LFSR, the snake module which took into account the snake size and growing longer whenever it hits an apple, direction module which is a very simple module implements the FSM which address the direction of movement, and the VGA which displayed this animation on the screen. We had to make modifications on our VGA driver. There were some other smaller modules such as clock reduce aka clock divider which the 50MHz clock into 25MHz. The way the snake was implemented is using an FSM that has four states: up, down, left, and right. The logic is simple. Basically whenever the snake is going down, you can only go left or right so therefore we used a 2-bit wire called direction to show this two possibilities. For example, we used 2'b00 to indicate left and 2'b01 to indicate right, and 2'b10, 2'b11 to remain in that specific states (in this case down). The same exact logic was applied to all of them. Below is the FSM machine the snake module. Also, the block diagram is given at the bottom

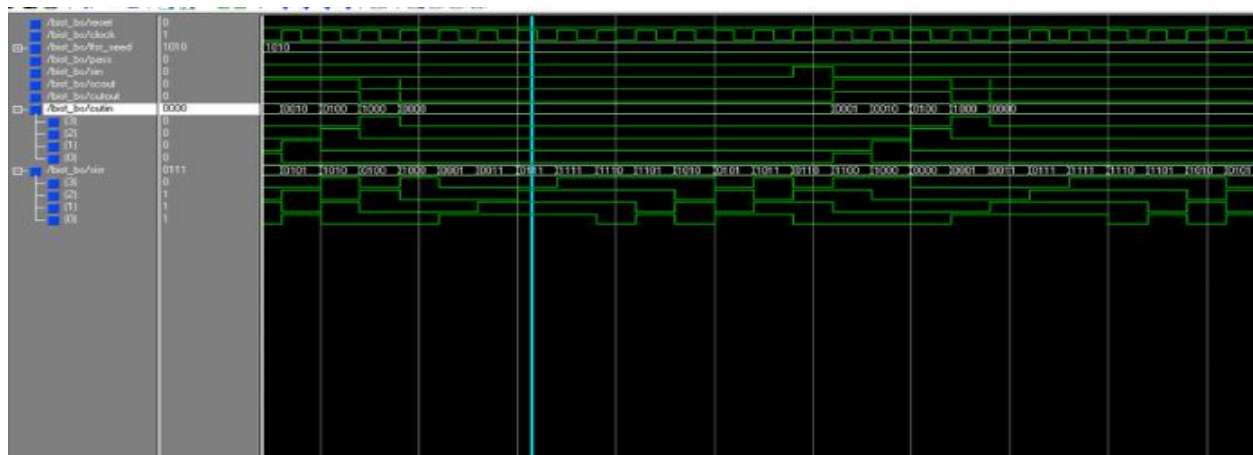


The final finished project was to have a snake moving in the direction of an apple and when it eats it should grow in size. Unfortunately, this wasn't accomplished and more of this will be discussed in the Problems Faced and Feedback section. The simulations below show the three main modules: Snake, direction abbreviated kb, and apple generator which instantiated the LFSR. The simulations are shown below

Main module (more like DE1_SoC) taking into account the randomly generated apple and the direction that's headed to

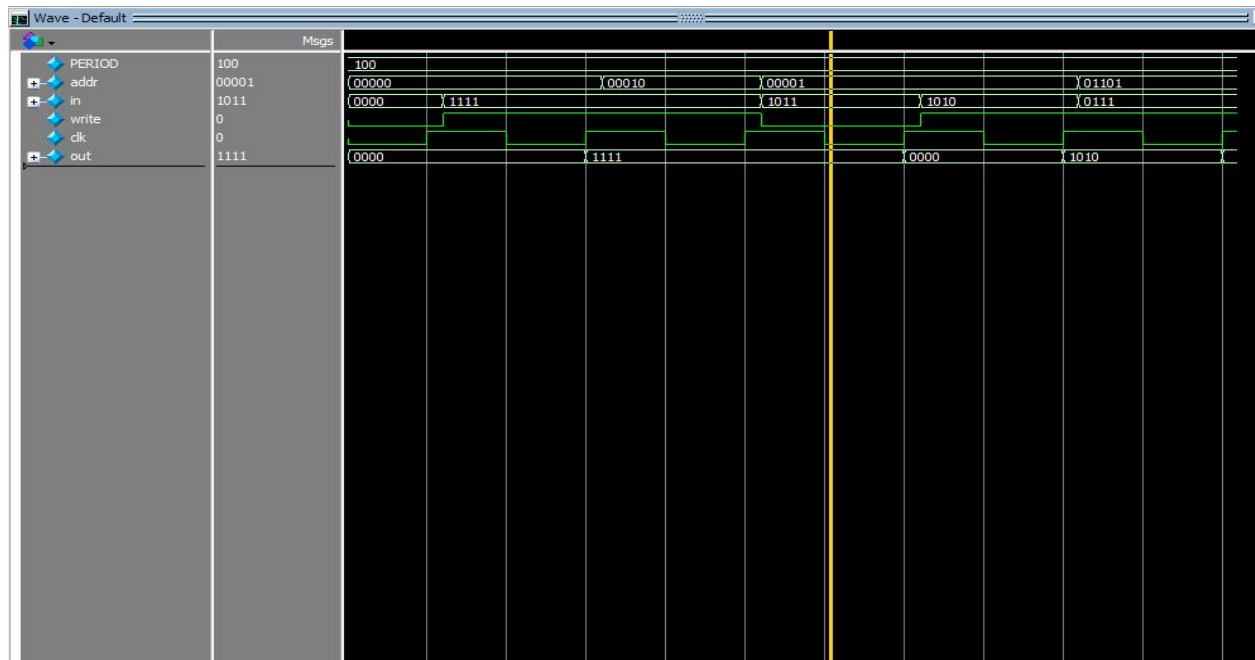


It just shows the apples in random spots on the screen. Very easier module



Direction:

Shows where it's headed to based on the simulation.



Problems Faced and Feedback:

This was a very challenging lab that we were not able to successfully complete it. I think our main issue was connecting between eating of the apple and snake on the same screen. The apple generator was quite easy. We were able to get it to have the apples all over but when we started implementing the snake module it wasn't working. The snake just shows dashes which are just moving the screen at a very fast pace even when we used a divider to slow down the VGA clock. The speed just won't change. Secondly, combining both modules won't work together. It was frustrating not able to figure out what was wrong. Even the simulation of the individual modules just gave us what we expected. The keyboard just worked fine. This lab took us over 80 hours on reading, planning, designing, coding, debugging and testing the code. I don't know any tip that I have for the lab other than just say start early and deeply thinking it through. I guess its better to have lectures dedicated to knowing how the VGA works. Other than the snake module not working, it took us many hours to understand what the VGA was really doing.