Coen 313 Digital Design Project

Done by: Mohammad Masoud Sharafat
Student ID: 40209284
Due date: June 17th, 2024
Professor: Dr. Otmane Ait Mohamed, PhD, ing.

“I certify that this submission is my original work and meets the Faculty's Expectations of Originality”

40209284
June 17th, 2024

1. Introduction and scope.

The scope of this project is to create a digital system that tracks and monitors the occupancy of a room, the system will work by having two separate sensors, one that detects when a person enters, and another that detects when a person leaves, the user will be able to input the maximum occupancy the room allows, up to 63 occupants, and once the current occupants equals the maximum allotted occupants, then an alarm will go off, once the alarm goes off, the system will not accept any more people to enter the room, and it will only accept requests to leave the room, though the system can be reset so that the system returns to its initial state.

2. High level description

In terms of the high level description, the system starts off with the current state equal to “000000”, this is also the value reset resets to. Then if the current state has a value equal to max occupancy, the output Z is set to 1, and the system waits for a reset or a decrement Y, but if current state is not equal to max occupancy, then the system checks the inputs X and Y, if X is equal to one, the system increments by one, and if Y is equal to one, the system decrements by one, thus if X is equal to Y, then there is no change in the output as both negate each other, and if X is greater than Y, then the current state is incremented by one, and if Y is greater than X, then the current state is decremented by one, this is done by a series of multiplexers to maximize speed. The whole system, excluding the asynchronous reset, is governed by the clock.

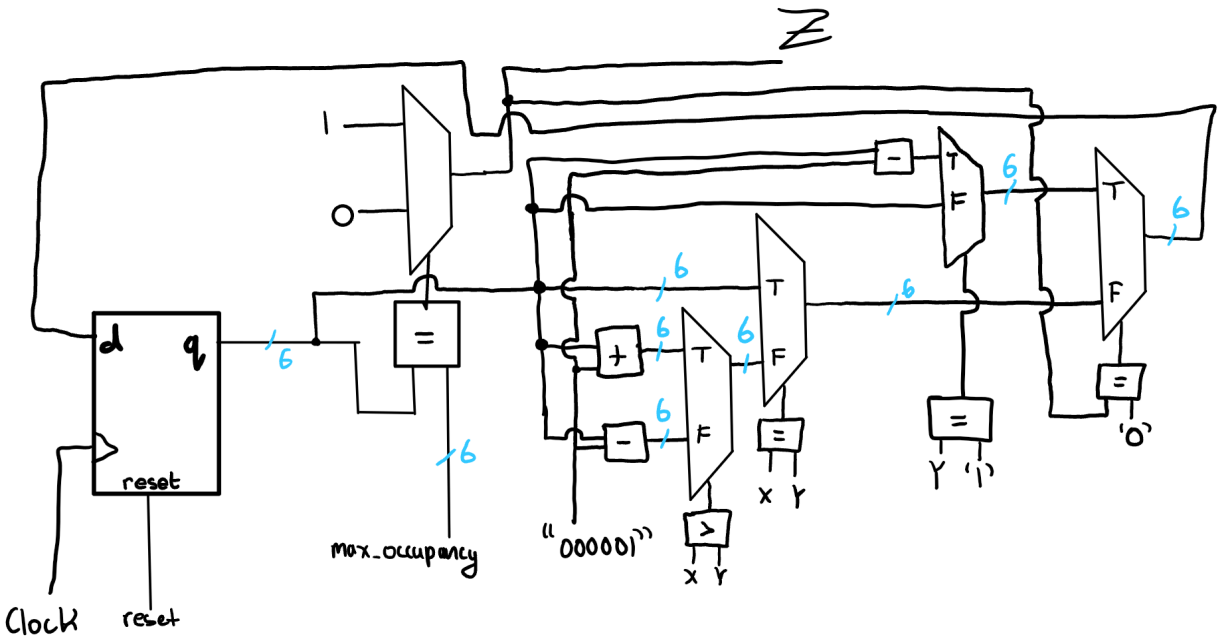

Figure 1. Conceptual Diagram of the system

3. Implementation and Model.

In terms of the implementation the entity included five inputs, X and Y which controlled incrementing and decrementing respectively, max_occupancy which controlled the maximum occupancy allowed, up to 2^6 bits, and then a clock and reset; one output Z was implemented, which equalled one when the occupants in the room equaled the maximum occupancy allowed, the entity is shown in the project.vhd file.

A process sensitive to the clock and reset was implemented, in this process, a variable named current_occupancy was implemented to hold the value of the current occupants. After the declaration, reset would be checked first, and if reset was equal to one, then current_occupancy and Z were equated to zero. If reset was not one, and the clock was at a rising edge, the circuit first checked if current_occupancy was equal to max_occupancy, if so, then Z = 1 and the system checked for the value of Y to see if a decrement is included, and if current_occupancy was not equal to max_occupancy, then the circuit checked for the values of X and Y, if X was equal to one, then current occupancy is increment, and if Y was equal to one, then current occupancy was decremented, the architecture is shown in the project.vhd file.

4. Simulation Results

A simulation was created using the test bench in project_tb.vhd. The test bench first check the functionality of the reset, which for the first two clock cycles correctly sets the max occupancy to “111111”. Then reset is set to zero, and max_occupancy, X, and Z are tested by setting max occupancy to “000011” or three and incrementing the current occupancy for 3 clock cycles, which as shown below, does properly set Z to one, as current occupancy is equal to maximum occupancy. The system is then reset, and after it is reset, max occupancy is set to “000010” or two, and this is followed by setting X to one for one clock cycle, then setting Y to one for one clock cycle, which results in the current occupancy being zero, to confirm this, X is set to one for two clock cycles, and since it took two clock cycles for Z to be set to one, we have confirmed the proper working of the decrementing Y. We then additionally set Y to one for one clock cycle, and observe Z being set to zero, this solidifies the proper working of Y and the responsiveness of Z to newer states. Thus this test confirmed the proper working of incrementing from X, decrementing from Y, resetting from reset, the proper function of clock cycles, the proper setting of a new max occupancy, and the proper output Z. This simulation ran for 2000 nanoseconds.

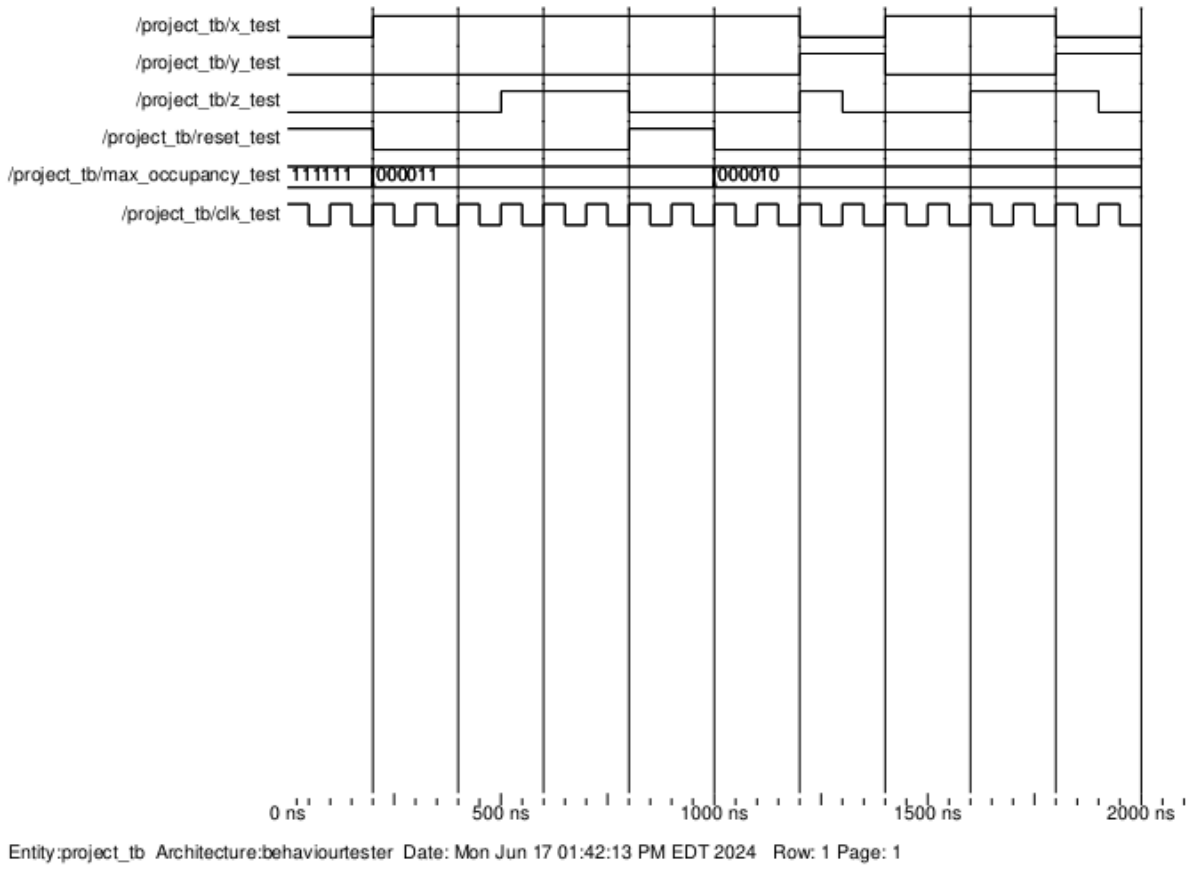

Figure 2. Simulation output, first 2000 Nanoseconds, simulated on ModelSim.

A second test is then followed to test the maxima, as in when the maximum occupancy is set to “111111” or sixty-three, and this is done by implementing a for loop that set’s x to one for sixty-three clock cycles, or until Z is equal to one, after which reset is set, and the test bench outputs a report stating that Z was successfully set to one, this simulation completed successfully, and is shown below.

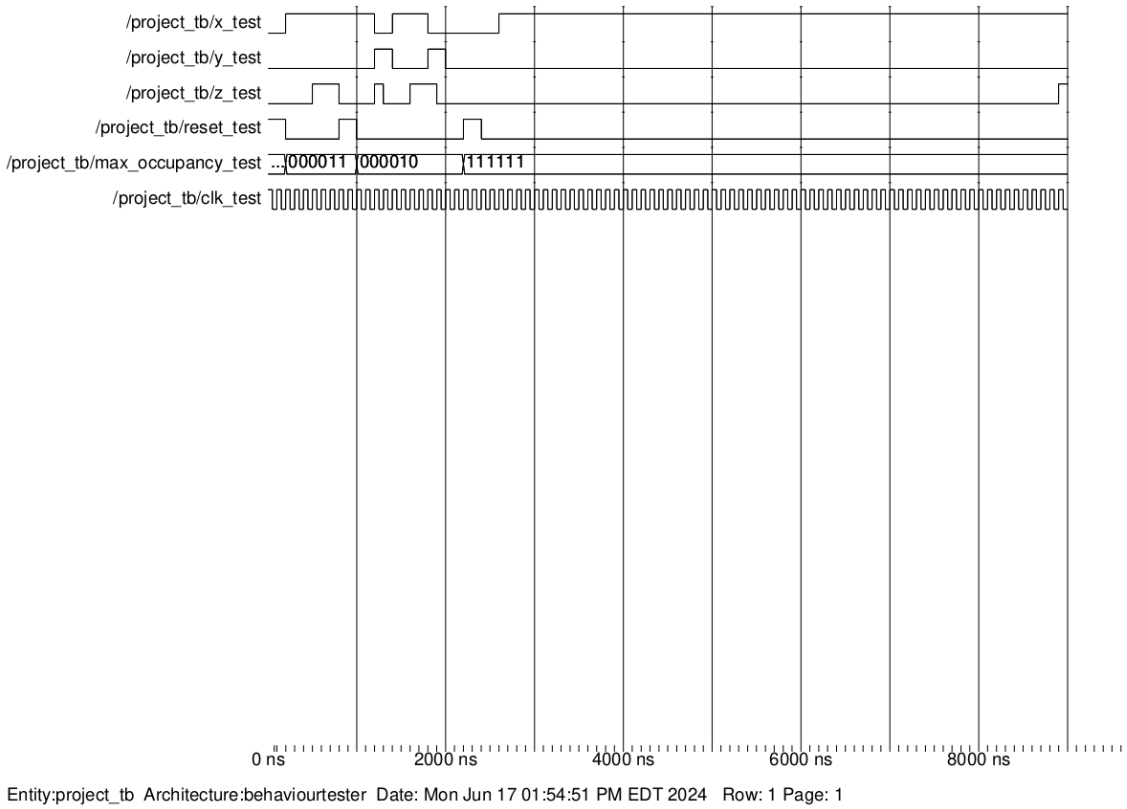

Figure 3. Simulation output, full range, simulated on ModelSim.

5. Synthesis , implementation, and RTL elaborated design results.

After the system is properly compiled and simulated, the system is synthesized using Vivado. The system was synthesized with no real warnings or errors, the only warning was due to the lack of an XDC file, which is unnecessary since this wont be implemented in an FPGA as stated in the project description, the errors from the implementation were also mainly due to the lack of an XDC file, the implemented design is shown below.

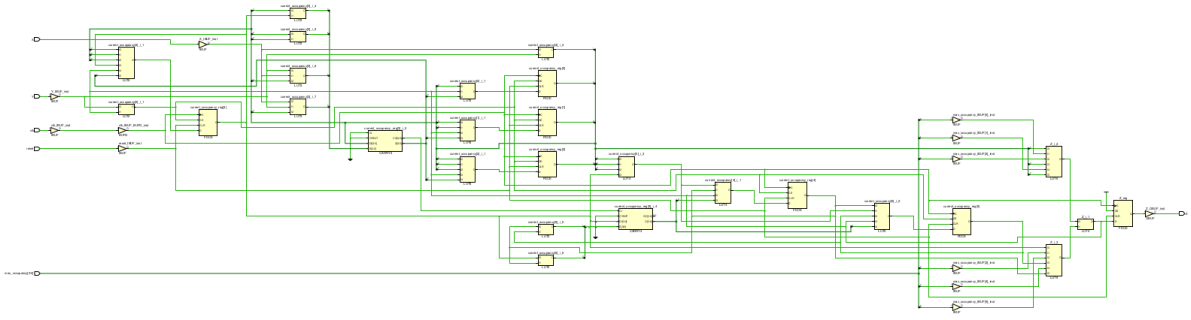

Figure 4. Implemented Design, Output from Vivado.

In terms of the RTL elaborated design, it is shown in the figure below.

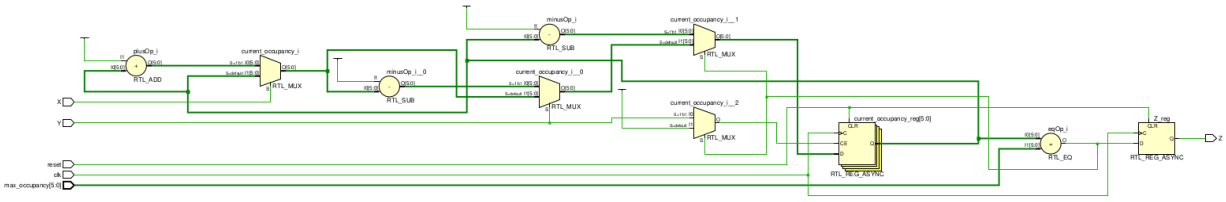

Figure 5. RTL Implemented Design, Output from Vivado.

As mentioned earlier, the Vivado synthesis and implementation log files only gave warning due to the lack of an XDC file, and generating a bitstream resulted in an error due to the lack of an XDC file, though this was an expected error, the implementation and synthesis logs are provided in a separate file due to their length, all files will be described in the read me.

6. Quality of design

In terms of the quality of the design, we can see the resource utilization in the table below, in terms of resource utilization, 16 Look up tables, 7 flipflops, 11 IO ports, and 1 BUFG are utilized, and as shown in the table below, very little of the LUTs and FFs of the FPGA are utilized, this is due to the relatively small size of the design compared to the xc7a100tcsg324-1 FPGA board's capabilities.

Utilization Post-Synthesis | **Post-Implementation**

Graph | **Table**

Resource	Utilization	Available	Utilization %
LUT	16	63400	0.03
FF	7	126800	0.01
IO	11	210	5.24
BUFG	1	32	3.13

Figure 8. Utilization Post-Implementation output from Vivado.

7. Conclusion

In conclusion, the system was programmed properly, resulting in a proper compilation, simulations were successful, and showed the proper working of all the functionality covered in the scope, and the system was properly synthesized and implemented into hardware.