Image 0 Output (Using segment of accuracy test bench to show results):

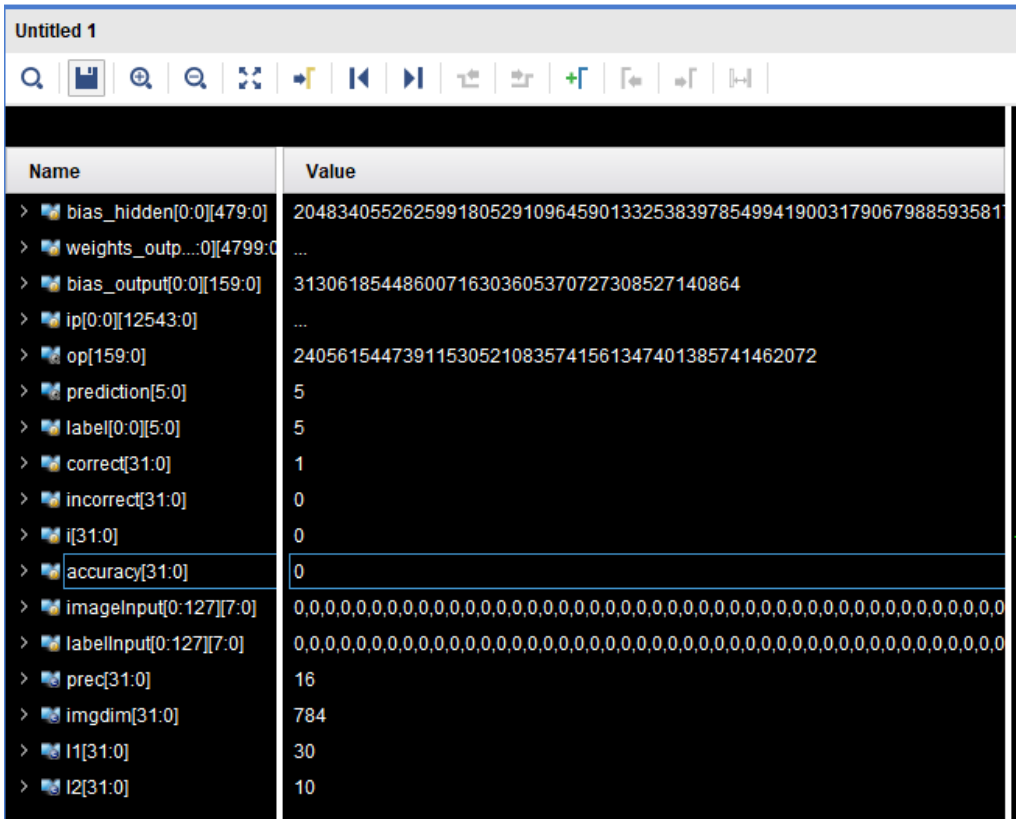
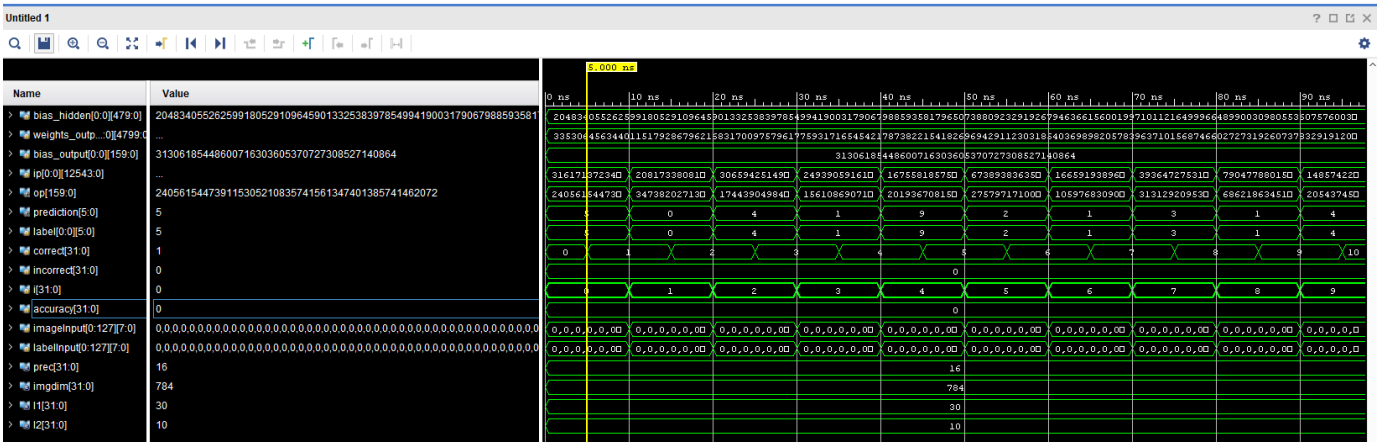


Image 0 Waveform:



- You can see that both the label and the prediction have the same value as well as the correct register incremented by 1 when this prediction was made. The matching values were 5 for prediction and 5 for label.

Image 1 Output:

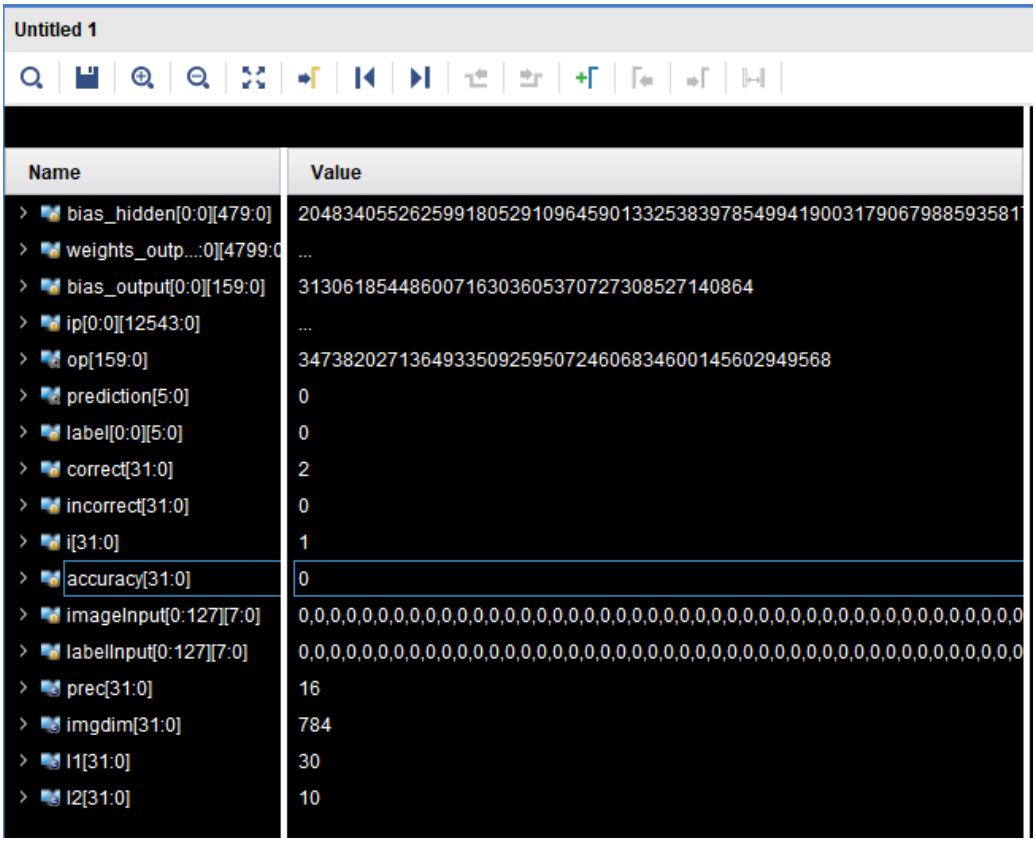
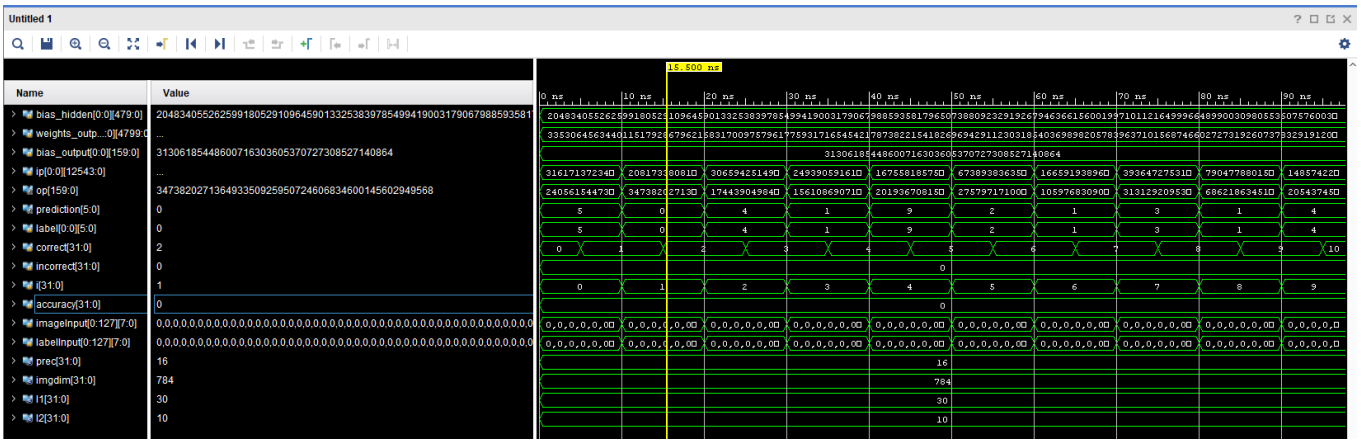


Image 1 Waveform:



- You can see that for the label and prediction registers here as well, they match and increment the correct register. The matching values are 0 for prediction and 0 for label.

Image 2 Output:

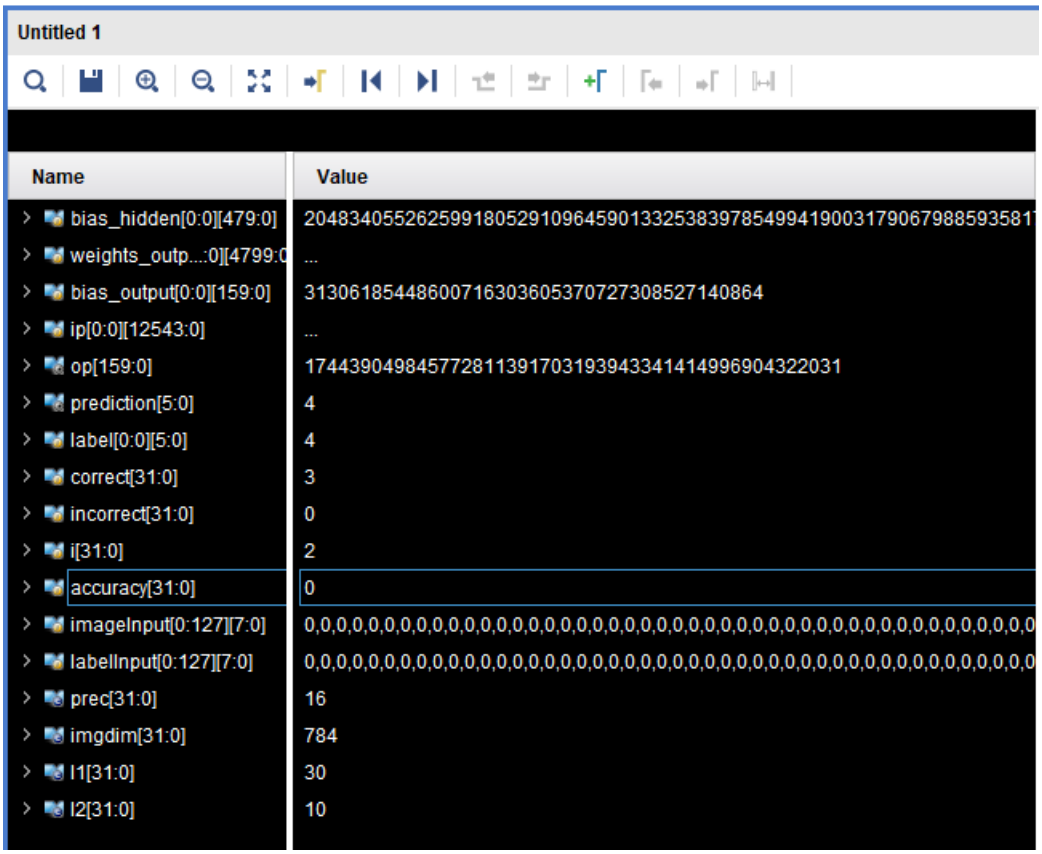
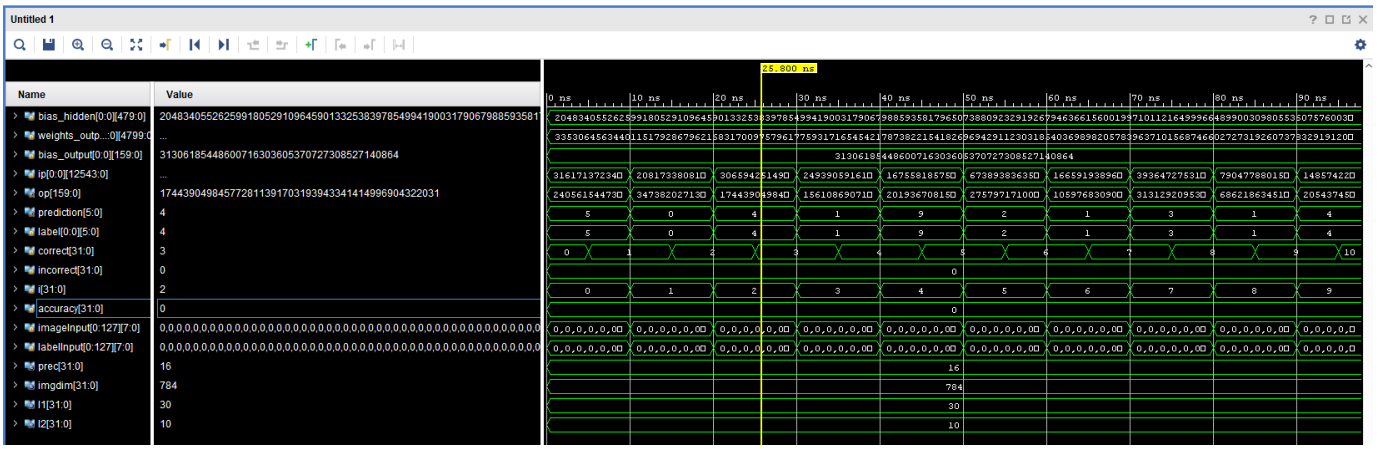
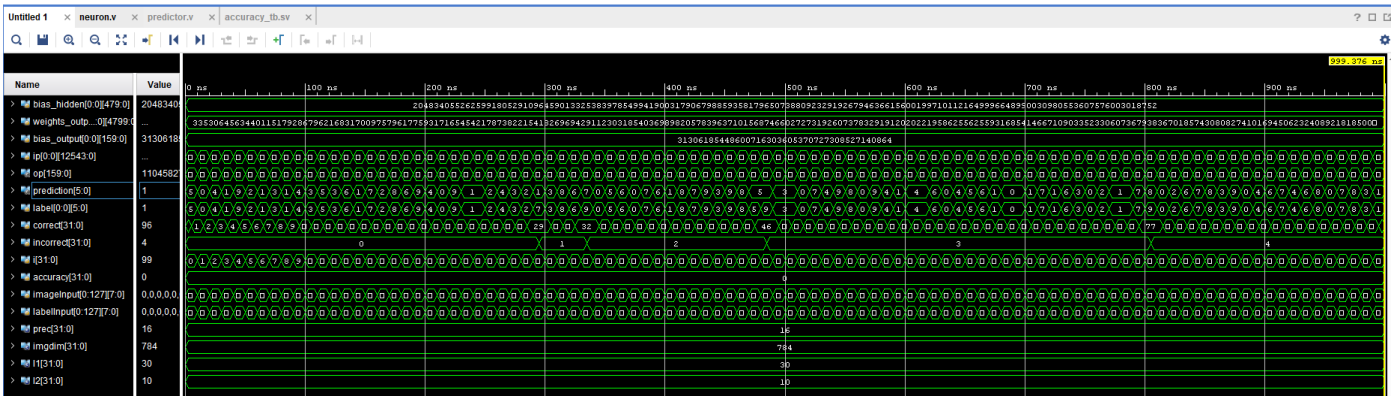


Image 2 Waveform:



- Image 2 also has a matching prediction value compared to the label value when ran through the neural network. Correct register gets incremented as well. The matching values are 4 for prediction and 4 for label.

Accuracy Test Bench full waveform:



Accuracy Results:

```
Time resolution is 1 ps
source accuracy_tb.tcl
# set curr_wave [current_wave_config]
# if { [string length $curr_wave] == 0 } {
#   if { [llength [get_objects]] > 0} {
#     add_wave /
#     set_property needs_save false [current_wave_config]
#   } else {
#      send_msg_id Add_Wave-1 WARNING "No top level signals found. Simulator will start without a wave window. I
#   }
# }
WARNING: [Wavedata 42-489] Can't add object "/accuracy_tb/weights_hidden" to the wave window because it has 3763
# run 1000ns
The accuracy is 96%
INFO: [USF-XSim-96] XSim completed. Design snapshot 'accuracy_tb_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:04 ; elapsed = 00:00:07 . Memory (MB): peak = 870.457 ; gain = 88.664
```

- As you can see, the accuracy simulation produces as result of 96%

Challenges and Implementation:

- There were lots of challenges when making this neural network. The first challenge was understanding how to modify certain wires based on a for loop.
- One big thing as well was understanding the two types of for loops we used in this project. One was a generate for loop using a genvar variable allowing us to identify wires the normal way. The other was a for loop within an always statement which required an integer variable as well as a different format in identifying specific wires.
- Understanding the layout of the network was also very important in successfully getting the result required.
- Another new challenge was the special way of overwriting parameters when instantiating modules.
- Lastly, I did have to do a different implementation in the accuracy test bench. We were instructed to use $sformatf to modify the file path strings, but my software wouldn't recognize it as a function. Because of this I used $sformat and had to store the string in an array of 8 wire static logic variables and then put those in the $readmemb function. I left the requested $sformatf option commented out to show how it would be implemented the normal way.