CPE 322 Simulation 2

Esther Shore

February 2024

*Clock\_24\_Hour\_structural.png* and *Clock\_24\_Hour\_behavioral.png* show the waveform output of the first testbench run with each of the structural and behavioral clock implementations. The simulations produced no differences, illustrating the correctness of the design over the range of inputs that are being simulated.

The first testbench *Clock\_24\_Hour\_tb1.v* is able to demonstrate the asynchronous functionality of the reset\_time and set\_time as well as the synchronous count from 23:59:55 to 00:00:05. We initially loaded the time\_out with 12:34:56 in our time\_in, then subsequently set the reset\_time bit to 1 while set\_time was still 1. This resulted in the clearing of time\_out, properly exhibiting that reset\_time takes precedence over set\_time. Then, the value of 23:59:55 was loaded and the manipulator bits set low so that counting ensued on the next rising edge of the clock. It was also ensured that these actions were executed not during a positive edge of the virtually generated clk signal to show that they are asynchronous. The clk signal was generated through a calculated toggle of a bit every half period. A delay of 10 periods would then allow our program to count up 10 seconds, reaching our desired 00:00:05.

*Clock\_24\_Hour\_structural.txt* and *Clock\_24\_Hour\_behavioral.txt* show the list output of the first testbench run with each of the structural and behavioral clock implementations. This shows the value of each variable as they change at different timestamps through the runtime of the testbench, clearly showing our load, reset, and count operations in addition to the steady clock signal.

The automatic testbench *Clock\_24\_Hour\_tb2.v* that exhaustively simulates and reports any discrepancies for the entire time-of-day count sequence displayed no differences for both implementations as well, further verifying our designs. This was helpful in identifying an accidental oversight in the carry over for the ones place of the hour, resulting in certain sequences in the count being incorrect, and subsequently fixing it until no differences were reported by the automatic testbench.