**Mini Project 2 Report**

The circuit described in this code is designed to drive an RGB LED using PWM signals to smoothly cycle through colors on the HSV color wheel, completing one full cycle every second. There are three key components of this project that are implemented in SystemVerilog: PWM signal generation, light fading, and color changes.

How this actually plays out in hardware is that the FPGA receives a periodic electrical signal called the clock signal, which drives all of its operations. The frequency of this signal determines the speed the LEDs fade in and out at, cycle through color, and is used to generate the PWM signal. The FPGA generates a PWM signal by rapidly switching the signal on and off at the rate of the duty cycle, which causes the LEDs to fade in and out and gives the visual of the color changing. In this circuit configuration the FPGA generates three PWM signals for each color channel so that they can be offset from each other. This is done in the PWM module. The LEDs duty cycle is set by the fade module, which uses a counter to cyclically dim and brighten the LED. Each of the fading LEDs is offset from each other by the color cycling module, which initialized them at different points to create a gradient effect between the lights. The top module then sends the signal to the correct LED pins to physically demonstrate the effect. .

Overall, the code implements a hardware system where each RGB LED color smoothly fades in and out using PWM, creating a dynamic lighting effect. Each color's fading is staggered, and the PWM signal ensures precise control over the brightness of the LEDs.

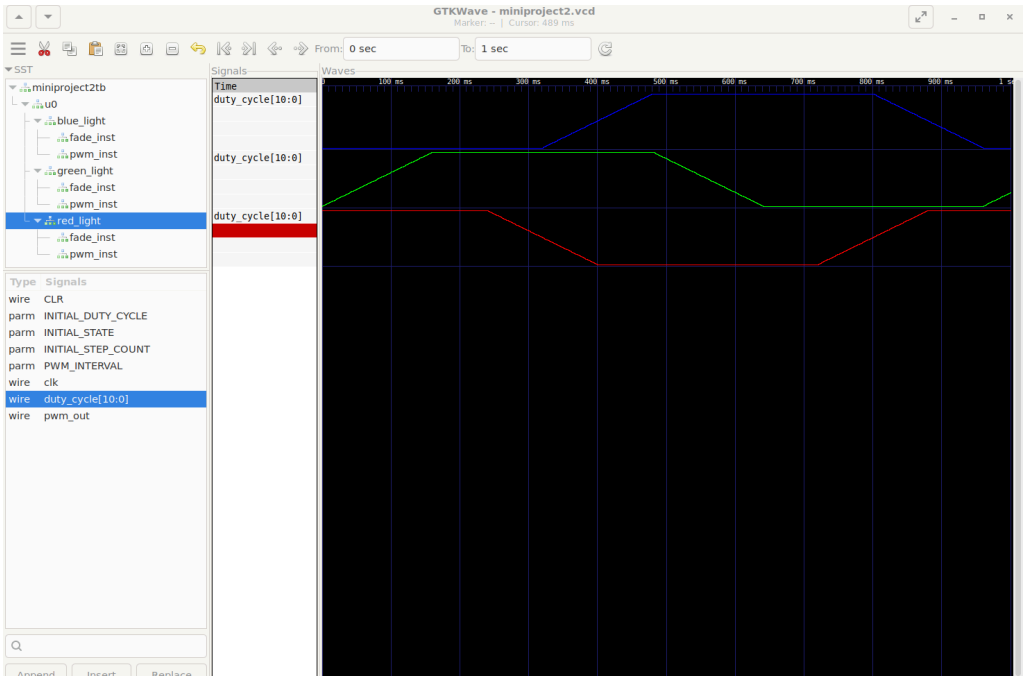

Fig 1: Waveforms showing the offset duty cycles of each of  the different colored lights.