**Objetivo**

Utilizando un LED RGB, Controlar la intensidad de encendido de cada color de tal manera que la luz resultante vaya mostrando los colores del arcoíris.

El máximo de salidas que tenga la tarjeta de desarrollo son los colores que se van a mostrar, por ejemplo, si se tiene 21 puertos de entrada/salida (E/S) pueden tener 7 colores diferentes. Para los colores rojo, verde y azul solo se requiere de un puerto por cada color (total tres puertos E/S)

**Introducción**

Para esta practica deberemos repasar varios conceptos vistos tanto en materias pasadas como en la materia de Diseño Digital VLSI. Algunos de estos conocimientos requieren saber el funcionamiento de las tarjetas FPGA, su arquitectura, y cómo programarlas utilizando lenguajes de descripción de hardware como en este caso VHDL. También se necesitará conocimiento de electrónica para realizar la conexión y configuración de periféricos en este caso, el LED RGB. Además del led se necesitará saber cómo usar los puertos de entrada/salida (E/S) en la FPGA para interactuar con el LED RGB y ajustar la intensidad de cada color.

Para la realización del código usaremos de apoyo los códigos proporcionados por el profesor en practicas anteriores como lo es el módulo PWM para controlar la intensidad de cada color del LED RGB. Para esto debemos saber cómo generar señales PWM en la FPGA para ajustar la intensidad de los canales rojo, verde y azul por separado. Para mostrar los diferentes colores en el led se deberá tener un conocimiento básico de la representación de colores en sistemas digitales, como el modelo RGB, y de cómo mapear los colores del arcoíris a los valores adecuados de los canales de color. En esta práctica se hará uso del software de Intel Quartus, para escribir, sintetizar y cargar el código en la tarjeta FPGA.

**Desarrollo**

Utilizando los módulos funcionales de prácticas anteriores (PWM y divisor de frecuencia) y el código base proporcionado, modificamos dicho código para hacer que un LED RGB mostrara los colores del arcoíris.

Comenzamos investigando a cerca del LED RGB que prácticamente es la unión de 3 leds de los colores básicos: rojo, verde y azul, cada uno de estos con su respectivo pin.

Ahora para implementar el LED RGB en la FPGA DE10-Lite es conveniente utilizar un Led RGB de cátodo común, esto para que en función de la señal PWM podamos variar los colores en el LED.

El código modificado es:

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_arith.all;

use ieee.std\_logic\_unsigned.all;

entity p\_2 is

Port(

clk: in std\_logic;

ledR: out std\_logic;

ledG: out std\_logic;

ledB: out std\_logic

);

end p\_2;

architecture arqLedRGB of p\_2 is

component divisor is

generic (N: integer:=24);

port(reloj:in std\_logic;

div\_reloj: out std\_logic

);

end component;

component pwm is

port(reloj\_pwm:in std\_logic;

D: in std\_logic\_vector (7 downto 0);

S: out std\_logic

);

end component;

signal relojPWM: std\_logic;

signal relojCiclo: std\_logic;

signal a1: std\_logic\_vector(7 downto 0) := x"00";

signal a2: std\_logic\_vector(7 downto 0) := x"00";

signal a3: std\_logic\_vector(7 downto 0) := x"00";

begin

D1: divisor generic map (10) port map (clk, relojPWM);

D2: divisor generic map (23) port map (clk, relojCiclo);

P1: pwm port map (relojPWM, a1, ledR);

P2: pwm port map (relojPWM, a2, ledG);

P3: pwm port map (relojPWM, a3, ledB);

process (relojCiclo)

variable Cuenta : integer range 0 to 255:=0;

begin

if relojCiclo = '1' and relojCiclo'event then

if cuenta = 0 then

a1 <= x"FF"; --Amarillo

a2 <= x"FF";

a3 <= x"00";

elsif cuenta = 5 then

a1 <= x"00"; --Verde

a2 <= x"FF";

a3 <= x"00";

elsif cuenta = 10 then

a1 <= x"00"; --Azul

a2 <= x"60";

a3 <= x"FF";

elsif cuenta = 15 then

a1 <= x"FF"; --Rojo

a2 <= x"00";

a3 <= x"00";

elsif cuenta = 20 then

a1 <= x"FF"; --violeta

a2 <= x"60";

a3 <= x"EF";

elsif cuenta = 25 then

a1 <= x"FF"; --naranja

a2 <= x"30";

a3 <= x"00";

elsif cuenta = 30 then

a1 <= x"FF"; --rosa

a2 <= x"60";

a3 <= x"60";

end if;

cuenta := cuenta+1;

if cuenta = 35 then

cuenta :=0;

end if;

end if;

end process;

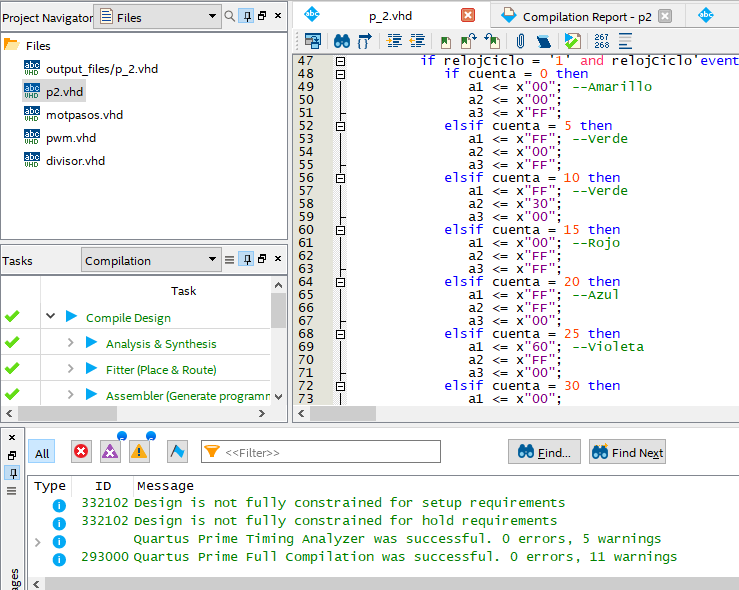
end arqLedRGB;

Las modificaciones en este código fueron varias y están resaltadas en color amarillo

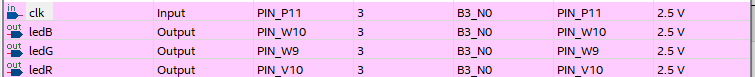
Primero se agrego un contador el cual cada que se completan ciertos números de ciclos cambia el color a mostrar. En este caso usamos los colores del arcoíris los cuales son: verde, amarillo, azul, morado, rojo, naranja y rosa. Además de esto se modificaron las señales de entrada del proyecto ya que se iba a trabajar con solo los leds RGB. Además de esto renombramos la variable reloj\_in a clk.

El contador nos ayuda a cambiar entre los diferentes colores que vamos a mostrar esperando a que se realicen 5 ciclos de la variable relojCiclo, al pasar estos 5 segundos se cambia el valor de las entradas del componente pwm, causando que se cambien los colores de los leds a los colores que elegimos anteriormente. Para saber las tonalidades de los colores se realizaron diferentes combinaciones en los valores del de cada led RGB.

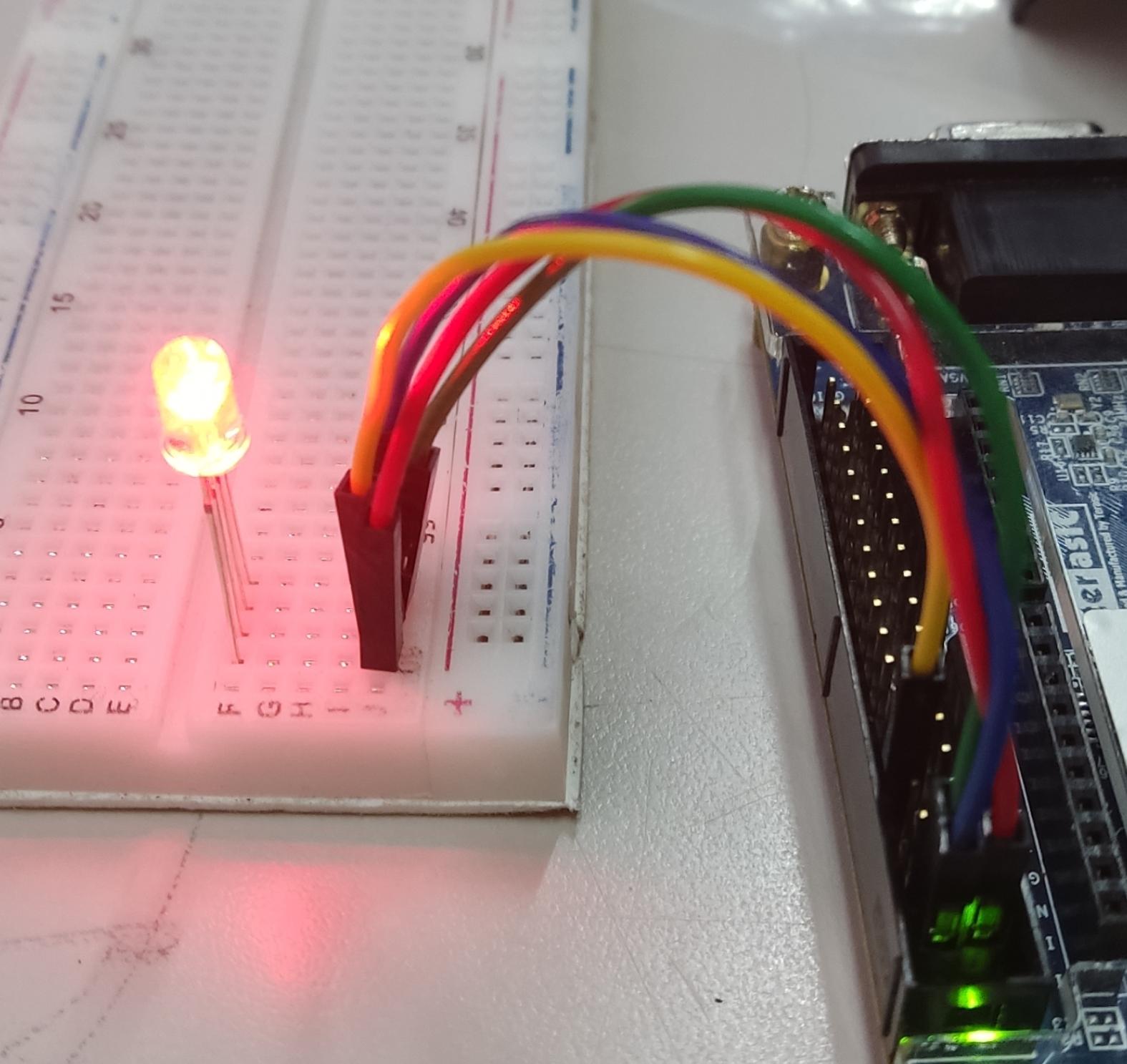
Prueba de compilación y ejecución



La configuración de los pines es la siguiente:



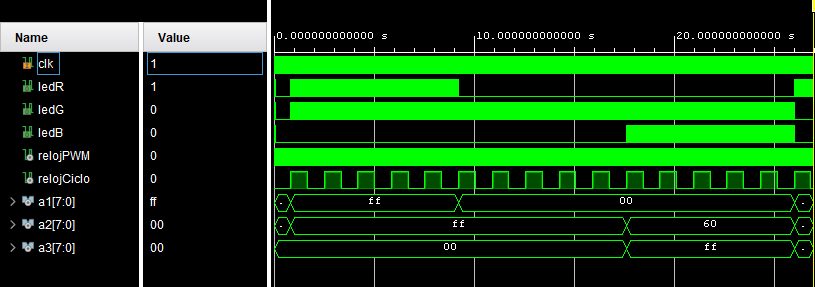
* Los ledR, LedG, LedB corresponden a las patitas del led RGB estos serán conectados a las entradas GPIO de la tarjeta de10lite en este caso los pines V10, W10 y W9 respectivamente
* CLK es el rejoj principal de 50Mhz.



Además de las conexiones a los pines se conectará el cátodo común a la tierra de la tarjeta para asegurar el funcionamiento del led.

Realizando la simulación

Obtenemos lo siguiente donde se puede observar los diferentes cambios en las tonalidades de las entradas RGB



El video mostrando el funcionamiento del LED es el siguiente:

<https://youtu.be/hoJt2PFVwSs?si=JPZ8YlT7H0V8tjUV>

**Bibliografía**

* Pedroni, V. A. (2010). *Circuit design and simulation with VHDL*.
* Terasic. (2017). DE10-Lite User Manual. <https://www.terasic.com.tw/cgi-bin/page/archive_download.pl?Language=China&No=1021&FID=a13a2782811152b477e60203d34b1baa>
* Jaramillo, D. M. (2022, 19 abril). LED RGB con arduino. *Programar fácil con Arduino*. <https://programarfacil.com/blog/arduino-blog/led-rgb/>