

# Generador de PWM personalizado para leds

**CAMILO GUTIERREZ<sup>1</sup>, RAIMUNDO ANDRADE<sup>1</sup>**

<sup>1</sup>Pontificia Universidad Católica de Chile (e-mail: primerintegrante@uc., segundointegrante@uc.cl)

SI Autorizo que mi proyecto (tal como ha sido entregado, sin nota ni comentarios de evaluación) sea publicado en un repositorio para pueda servir de guía y ser mejorado en proyectos de futuros estudiantes.

Este proyecto ha sido desarrollado bajo el curso IEE2463: Sistemas Electrónicos Programables.

• **ABSTRACT** El proyecto tiene como proposito la generacion de señales PWM en los 3 primeros leds comunes, y tambien controlar uno de los leds RGB con 3 señales PWM. Estas señales PWM podran ser generadas mediante 3 operaciones disponibles: Multiplicacion y division por un valor que estara representado en binario en los switches de la zybo, la tercera operacion corresponderia a la modificacion del duty cycle que por defecto estara establecido en 50%, el cual tambien podra ser modificado con los switches, usando una resolucio de valores 15, por lo cual si se selecciona en binario el valor 8, sera cercano al 50%. El boton 3 tambien tendra una funcion, dado que seleccionara los leds a modificar y sobre cual se aplicaran las operaciones, por lo cual al inicializarse siempre partira seleccionado el led0, por lo cual se podran hacer las operaciones directamente sobre este led, y si se aprieta el boton 3, sera seleccionado el led1 en el cual ahora se aplicarian las operaciones ligadas a los botones, y así continuamente saltandose el led3 ya que este tendra una unica funcion de verificacion de operacion, el cual se iluminara cada vez que se realizia una de las oepaciones mencionadas por lo que despues del led2 se movera al led RGB, y si se aprieta una vez mas el boton 3 volvera al led 0.

En esta parte escriba un resumen de su proyecto indicando: i) que hace su proyecto, ii) que elementos y tecnologías utilizó para lograr hacer funcionar su proyecto, iii) cuales fueron sus resultados. Este resumen no dbe contener más de 250 Palabras.

## • INDEX TERMS

PWM, AXI-lite, Custom Slave IP, Duty Cycle, Frequency modification. Escriba las palabras claves con las cuales su proyecto podría ser encontrado en internet. Por ejemplo VHDL, maquina de estados, memoria, etc.

## I. INSTRUCCIONES

Por favor borre esta sección antes de entregar su trabajo. Para entregar su informe debe:

- El documento total **no puede superar las 8 páginas**, incluyendo las referencias.
- Completar TODA la información del encabezado de este documento
  - Fecha.
  - Titulo del proyecto
  - Nombre de ambos integrantes
  - Emails de ambos integrantes
  - Responder si desea permitir que su trabajo quede disponible para otros.
  - Completar el abstract
  - Completar los Index terms
- Completar las secciones "Arquitectura de Hardware",

Actividades Realizadas", "Resultados de Simulacion", "Resultados de Implementación", "Conclusiones", "Trabajos Futuros" y "Referencias".

- El formato del informe es fijo, ud. no debe agregar, mezclar o eliminar secciones.
- La nota del informe será igual al total de puntos obtenido más un punto base.

## II. ARQUITECTURA DE HARDWARE (1 PUNTO)

**L**A estructura consiste en la instanciación de un generador de PWM el cual corresponde a un component que mediante los botones 0, 1 y 2 puede realizar la operación de multiplicar y dividir la frecuencia(boton 1 y 2) y modificar el duty cycle. El valor modificador para estas operaciones se entrega mediante la representación en binaria con los 4 switches correspondientes, siendo los valores disponibles

para multiplicar o dividir la frecuencia de 0 hasta 16, y la resolución de modificación de duty cycle de 16 posibles valores, por lo cual el valor 8 genera una señal con duty cycle aproximado a 50% la salida de este componente es una señal de naturaleza std logic, por lo cual al instanciarse 3 veces se realiza con el propósito de generar 3 señales PWM para 3 leds y además siendo estas 3 señales las mismas que ingresaran al led RGB, por lo cual la señal de led0 también es la misma señal que la de la componente R, la señal led1 análogamente para la componente G y led 2 para la componente B. La asignación a cada uno de estos leds se hace mediante el componente selector de led, el cual al recibir de entradas las señales pwm además de la dirección de operación actual (Que led se esta manipulando actualmente), asigna las señales pwm a sus leds correspondientes pero solo se activaran las seleccionadas mediante la dirección recibida de la memoria de direcciones que almacena las 4 posibles direcciones correspondientes a Led0, Led1, Led2 y LedRGB.

El componente operador tiene una salida que se interconectan cuando se instancian los 3 y que a su vez esta conectada al led3, esta salida enciende el led cuando cualquiera de las operaciones realizadas en alguno de los leds se completa.

La memoria de direcciones así mismo es una creación de memoria RAM similar a lo que se realizó en la ayudantía 4 del curso pero modificada para que cuando se apriete el botón 3 entregue la siguiente dirección a la salida, recorriendo su almacenamiento de 4 direcciones repetidamente. Esta memoria recibe la data de direcciones mediante comunicación AXI la cual es entregada por un ATG operando en test mode maestro, y entrega las direcciones a la memoria que esta definida como esclavo y en AXI lite.

La dirección entregada por esta memoria no solo se utiliza con el selector de led, sino también para enlazar correctamente los botones 0, 1 y 2 y los switches con el generador correspondiente según el led seleccionado.

Esta estructura se puede apreciar en el siguiente diagrama de bloques a continuación.

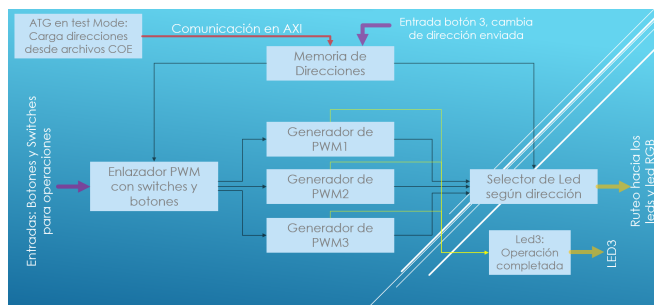


FIGURE 1: Diagrama de bloques de estructura funcionamiento

### III. ACTIVIDADES REALIZADAS (1.5 PUNTOS)

Describe el nivel de avance que obtuvo en cada una de las 10 actividades planteadas. Sea breve y guíese por el siguiente formato:

AO1 (100%): *Descripción: Descripción:* El código cumple con las funciones que habíamos planteado en nuestro proyecto las cuales son: Generar señales PWM manipulables mediante operaciones y representarlas en los leds y led RGB.

De acuerdo a la arquitectura presentada en la sección II, esto se implementó en los distintos bloques, y se hicieron bastantes package IP con entitis en su interior. El package IP *Chooser\_comps* esta constituido por 3 components similares que tienen como entradas las mismas entradas que el entity que describe el IP, pero que en su interior cada uno genera una distinta salida de valores constantes por lo cual si bien son similares no son iguales, estas components producen unas señales que luego en la entity se conectan a unas señales que dependiendo del estado entregado por la memoria ram de direcciones entregara una salida con los valores provenientes de su component. Por lo cual el objetivo esta logrado ya que se cumplen 3 components en una entity.

AO2 (100%): Se implementaron 4 packages y un 5to package asociado a la memoria RAM, que está basada en lo propuesto en la ayudantía 4.

AO3:(50%) Se utilizó comunicación AXI para comunicar un IP-core esclavo propio con un ATG maestro en test mode.

AC1: (100%) La operación del proyecto consiste en una máquina de Mealy pues la salida del sistema depende del estado actual y la entrada. Los estados de la máquina generada corresponden a los valores que adquiere selector\_led en función de la entrada en BTN3 de la zybo. Como se muestra en la figura X, el estado 00 corresponde a la configuración de la PWM0 que se visualiza en LED0, el estado 01 corresponde a la configuración de la PWM1 que se visualiza en LED1, el estado 10 corresponde a la configuración de de la PWM2 que se visualiza en LED2 y finalmente el estado 11 corresponde a la visualización de las PWM antes configuradas pero ahora en los canales del LedRGB respectivamente.

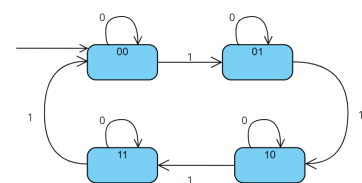


FIGURE 2: Máquina de estados de Mealy del sistema

AC2:(100%) Se utilizaron los 4 switches ya que fueron utilizados para representar un valor binario entre 0 y 16, los 4 botones ya que 3 corresponden a las operaciones de multiplicación, división, duty cycle y el cuarto boton utilizado para cambiar de estado moviéndose de leds. Se utilizaron 5 leds, 4 leds comunes y uno RGB, 3 de los leds comunes correspondientes a 3 señales PWM, el cuarto común con la configuración de parpadear cuando se complete una operación y el led RGB utilizando las mismas 3 señales PWM para sus 3 canales.

AC3: (100%) Con respecto a los operadores, en los lógicos se utilizaron tales como "and", "or" y "not". También se us-

aron operadores de asignación, como "<=" para la mayorías de los ip, y ">=" en map port de la RAM y de la instanciación de los componentes en el enlazador de PWM con botones y switches, y se hizo uso de ":" para asignar valore iniciales a variables.

Los atributos de señal utilizados fueron "rising edge", también se usaron los atributos como:"range",.

AC4:(100%) Se utilizia en diferentes partes como en Selector de led, chooser o RAM en el cual se hace uso de variables y de "if", en los cuales una variable como por ejemplo "countB" en el package de ram, es una variable que va cambiando con cada aretar de boton y en el codigo se quencial se elige que valor enviar al output.

AC5:(100%) Se logro dado que se hizo uso de codigo secuencial al utilizar "if", "elsif", secuencial se uso nuevamente en los diferentes bloques de uso de condicionales como en la ram, "selector" y "Chooser", y tambien concurrente en diferentes ocasiones como al hacer el mapeo de puertos o al asignar las señales de salida de "selector" hacia los leds.

AC6:(0%) No se logro ya que no se hizo uso de ninguna funcion creada ni procedures.

AC7:(100%) Se logro ya que se implemento el led RGB y funciona correctamente, el cual nunca fue presentado en el curso.

#### IV. RESULTADOS DE SIMULACIÓN (3 PUNTOS)

Muestre su simulación post implementación de alguno de sus procesos con el fin de:

- (0.75puntos) Identificar los retardos (delays) que ocurren debido a la implementación real y que no existen en la simulación de comportamiento.

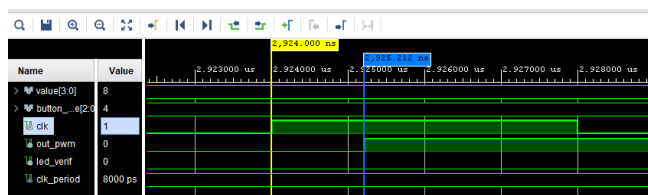


FIGURE 3: Retardo entre clk y señal actualizada

- (0.75puntos) La diferencia entre la actualización de señales y variables.
- (2puntos) Mostrar la correcta ejecución de las transacciones AXI Lite y AXI Full. Identifique claramente los handshake asociados a cada transaccion.

#### V. RESULTADOS IMPLEMENTACIÓN (0.1 PUNTOS)

La implementación fue exitosa finalmente ya que funciono correctamente como se esperaba el proyecto, pudiendo controlar las señales PWM y moverse entre los leds. Hubo distintos tipos de complicaciones, como por ejemplo debido

al hacer uso de varias IPs que nunca se logro descubrir la razon sobre el porque no se sobrescribían bien.

Otro tipo de complicación fue respecto a warnings con respecto a direcciones de registros de esclavos del AXI, lo cual tampoco pudimos entender el porque de esta advertencia ya que, se hizo de forma similar a lo visto en la ayudantia 4 y no presento ningun error en el funcionamiento del proyecto.

#### VI. TRABAJOS FUTUROS (0.2)

Dado que si bien logramos nuestros objetivos de que la taret Zybo funcionara e hiciera lo que quisiéramos, no se logro completar requisitos fundamentales como el uso de AXI full y una correcta implementacion de text bench de AXI, por lo cual para trabajos futuros se deberia como minimo alcanzar estos objetivos.

Como objetivos extra, se piensa en la mejora o el adaptar este funcionamiento para otros tipos de señales, como por ejemplo conectar un periférico que actué como parlante y trabajar con señales audible de forma que sea posible escuchar los efectos de modificación de las señales auditivas con operaciones similares a las trabajadas en este proyecto, también incluso en quizás realizar una modulación AM de sonido con frecuencias mas altas establecidas en otros canales.

#### REFERENCES

- [1] Ayudantía 1,2, 3 y 4, Curso IEE 2463, pontifica Universidad Catolica de Chile, 2023.
- [2] Laboratorio 3 y 4, Curso IEE 2463, pontifica Universidad Catolica de Chile, 2023.

...