

Semáforo Multi Modo

MARCO RAMIREZ¹, JOAQUÍN ERRANDONEA¹

¹Pontificia Universidad Católica de Chile (e-mail: maramirez4@uc.cl, jaerrandonea@uc.cl)

(SI/NO (POR FAVOR ESCOGER SI o NO) 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 presente proyecto se trata de un semáforo de 4 luces (verde, amarillo, rojo y viraje) representado en la tarjeta Zybo Z7-10. El led RGB se encarga de emitir las luces verde, amarillo y rojo, y un led genérico (comúnmente denominado led3) se encarga de indicar la señal de viraje. Cada switch representa un modo distinto al *default* del funcionamiento del semáforo. Los botones son usados para el modo manual (o de operador). Dado que se tienen 4 botones, cada uno manipula uno de los 3 estados (verde, amarillo y rojo) y otro incide directamente en habilitar el viraje sin importar el estado de la máquina. Se utilizaron 3 bloques IP Core: un bloque que toma la responsabilidad de leer las interacciones del usuario con los switches y botones (técnicamente llamado *interactions*), una máquina de estados *state machine* que cicla el semáforo según la luz que debía ser prendida y un driver que instruye a los leds cuáles debían encenderse y que colores debe tomar el RGB. Además de estos bloques, se utilizaron 2 generadores de tráfico AXI: uno en Test Mode y otro en Advanced Mode. El sistema no pudo ser implementado en la tarjeta Zybo por errores de implementación y compatibilidad entre bloques.

INDEX TERMS Vivado, VHDL, maquina de estados, semáforo, AXI.

I. ARQUITECTURA DE HARDWARE (1 PUNTO)

La estructura consiste en 6 bloques que funcionan a partir del mismo reloj: 4 IP Cores creados por nosotros y 2 generadores de tráfico AXI (uno en Test Mode y otro en Advanced Mode que debido a problemas de implementacion debió ser reemplazado por una PWM en). El ATG en Test Mode entrega datos al IP Core llamado "interactions" (axi en slave mode) que, a partir de lo entregado por el ATG y de los switches y botones apretados, define una salida de 8 bits enviada al bloque "selector". La máquina de estados a su vez recibe el estado actual (en forma de 2 bits) y entrega como salida el estado siguiente y un bit que indica si hay viraje o no. Estas salidas también van hacia el bloque "selector" como entradas. El selector entrega 3 salidas: una de estas (el estado actual) es realimentada a la máquina de estados y las otras 2 (leds por prender y el viraje) son entregadas al bloque "led display", que termina encendiendo los leds.

II. 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 (15%): Se logró desarrollar el código, pero sin bitstream.

AO2 (15%): Se logró desarrollar el código, pero no se pudo

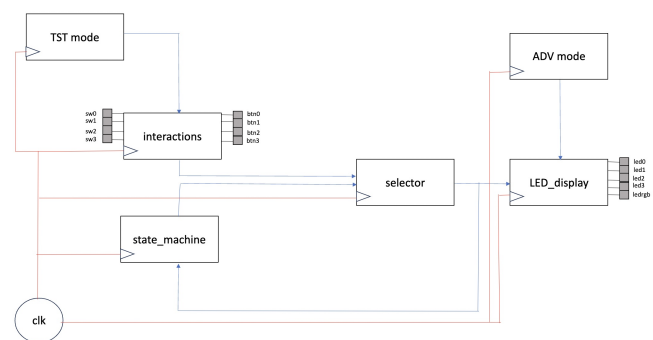


FIGURE 1: Diagrama de hardware

obtener su bitstream.

AO3 (5%): Se logró desarrollar el código para el AXI- lite, pero no se pudo obtener su bitstream.

AC1: (10%): Se logró desarrollar el código, pero no se pudo obtener su bitstream.

AC2: (10%): Se logró desarrollar el código, pero no se pudo obtener su bitstream.

AC3: (0%): No lograda.

AC4: (10%): Se logró desarrollar el código, pero no se pudo obtener su bitstream.

AC5: (10%): Se logró desarrollar el código, pero no se pudo obtener su bitstream.

AC6: (0%): No lograda.

AC7: (0%): No lograda.

III. RESULTADOS DE SIMULACIÓN (3 PUNTOS)

No se pudo realizar la implementación, por lo que no se pudo realizar una simulación del diseño post implementado.

IV. RESULTADOS IMPLEMENTACIÓN (0.1 PUNTOS)

Describe brevemente, en no más de dos párrafos, como le resultó su implementación. Los principales problemas que tuvo, cuales pudo resolver y cuales no.

V. TRABAJOS FUTUROS (0.2)

Este proyecto pretende servir como ejemplo de una máquina de estados, por lo que podría usarse para introducir al concepto de una máquina de estados tipo Mealy (sujeta a bucles) y también como punto de inicio para proyectos que trabajen con ciclos.

...