Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería Eléctrica
IEE2463 Sistemas Electrónicos Programables

Ayudantía 05 2025-2

ZYBO Z7-10 - AXI Full y RGB driver

Ayudante: David Rodríguez darodriguez8@uc.cl

Prof. Dr.-Ing. Félix Rojas - felix.rojas@uc.cl

---

1. Objetivos de la Ayudantía

- Ejercitar protocolo AXI full
- Uso de AXI ATG y archivos .coe
- Profundizar la creación de IPCores con puerto AXI
- Generación de señal PWM y Manejar LED RGB

2. Actividades previas a la Ayudantía

Para el desarrollo de esta actividad, **no se utilizará el proyecto base ejecutado en el video de la ayudantía 05, pero si se va a hacer uso de los conceptos y la lógica para generar la señal PWM, por lo que haber visto este último es necesario para comprender en su totalidad el funcionamiento de esta actividad.**

2.1. Pasos a seguir

- Cree un IPCore con puerto AXI full configurado en slave mode y un Memory Size de 64 Bytes. Es decir que la memoria va a soportar 16 datos de 32 bits (capacidad para almacenar la ram):

$$cantdad\_datos = \frac{64 \cdot 8}{32} = 16$$

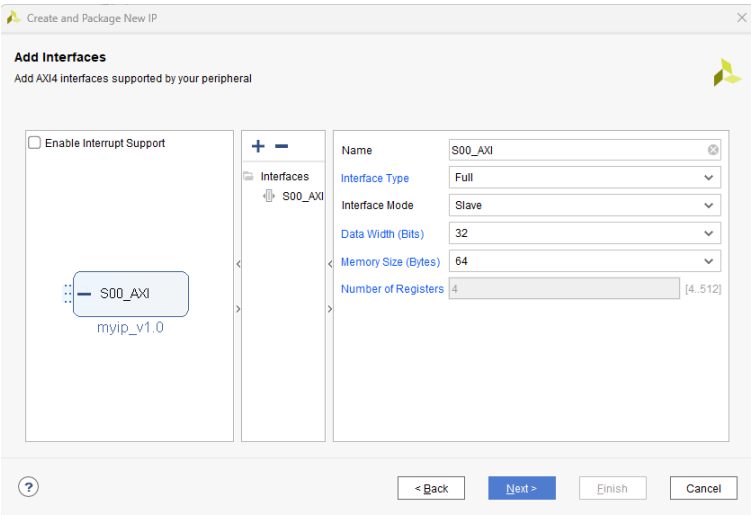

Figura 1: *IPCore AXI Full*

- Se abrirá un proyecto nuevo asociado al IPCore, incluya en la fuente de menor jerarquía el archivo .VHD LED_DRIVER_FULL_v1_0_S00_AXI, en la fuente de mayor jerarquía incluya LED_DRIVER_FULL_v1_0. Finalmente, empaquete el IPCore e incorpórelo desde el diagrama de bloques *add IP*.
- Incorpore un bloque AXI Traffic Generator, configurado en Advanced Mode, protocolo AXI4 con data width 32 bits:

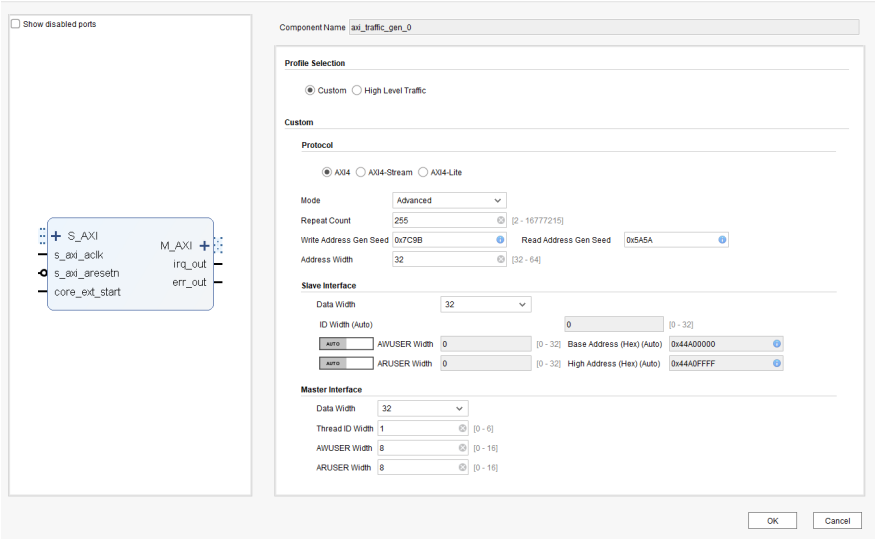

Figura 2: *AXI ATG advanced*

- Incorpore otro bloque AXI Traffic Generator, configurado en Test Mode, protocolo

AXI lite. Añada los archivo .COE de control, dirección, datos y mascara disponibles en CANVAS.

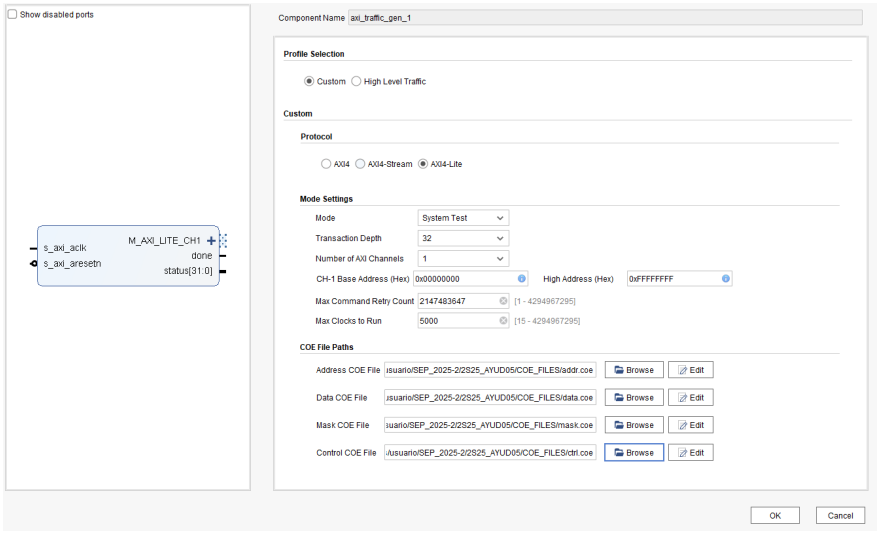

Figura 3: *AXI ATG Test Mode*

- Genere Run block connection automation, se le van a generar dos cloking wizzard, deje sólo el que alimenta el *slowest_sync_clk* y **conecte clk del IPCore al clk de la zybo**

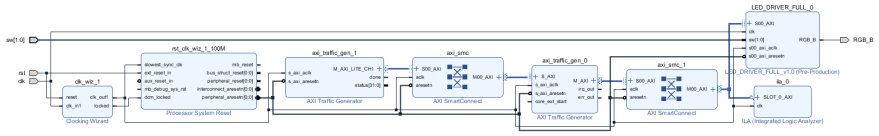

Figura 4: *Diagrama de bloques Final*

- **Verificar la correcta asignación de las direcciones de bloques AXI** en *Address Editor*:

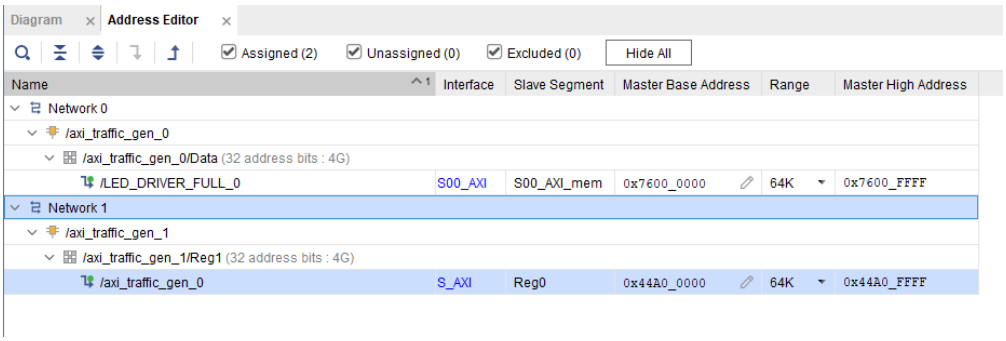

Figura 5: *Direcciones*

- Incluir archivo *Constraints* asociado a la tarjeta y editarlo de acuerdo al *Block Design* desarrollado.
- Incorpore un bloque ILA *MonitorType* AXI para medir y validar la correcta transacción de datos a través de AXI entre ATG y LED_DIVER_FULL como se puede ver en la **figura4**
- Generar el *HDL Wrapper* del *Block Design* y luego el bitstream, para cargarlo a su tarjeta Zybo y probar su funcionamiento.

Resultados

A continuación se presenta la transacción de los datos correspondiente al número que maneja la intensidad del LED_B

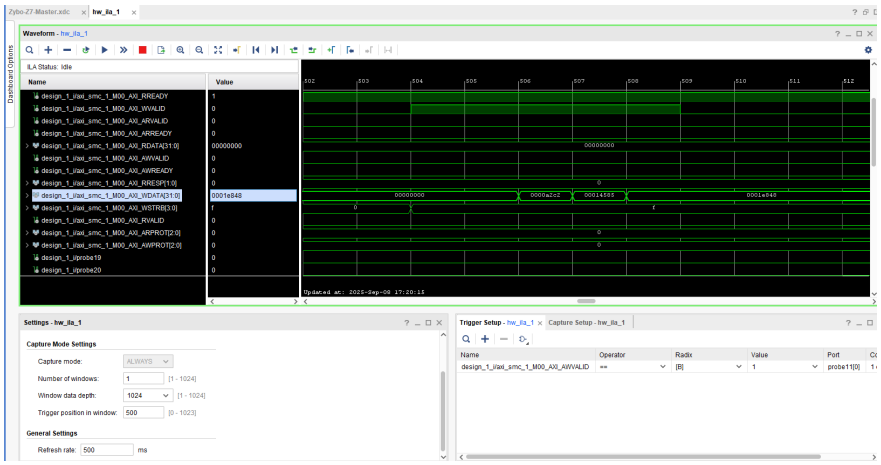

Figura 6: *ILA*

3. Actividades durante la Ayudantía

En el ejercicio propuesto de esta ayudantía deberá modificar el IPCore para generar tres colores, específicamente los colores **Yellow**, **Light Blue** y**Purple** según la siguiente tabla:

Color Chart	R	G	B	Color Name
	0	0	0	Black
	255	255	255	White
	224	224	224	Light Gray
	128	128	128	Gray
	64	64	64	Dark Gray
	255	0	0	Red
	255	96	208	Pink
	160	32	255	Purple
	80	208	255	Light Blue
	0	32	255	Blue
	96	255	128	Yellow-Green
	0	192	0	Green
	255	224	32	Yellow
	255	160	16	Orange
	160	128	96	Brown
	255	208	160	Pale Pink

Figura 7: *Tabla de valores RGB*

- Por ejemplo para crear el color morado se necesitan los siguientes ciclos de trabajo $\alpha_{rojo} = \frac{160}{255}$, $\alpha_{verde} = \frac{96}{255}$, $\alpha_{azul} = \frac{255}{255}$ en cada LED correspondiente. Tenga en cuenta que la amplitud del contador es el dato que se manda por AXI y genera el ciclo de trabajo necesario para el color que se quiere generar.
  También le puede ser útil:

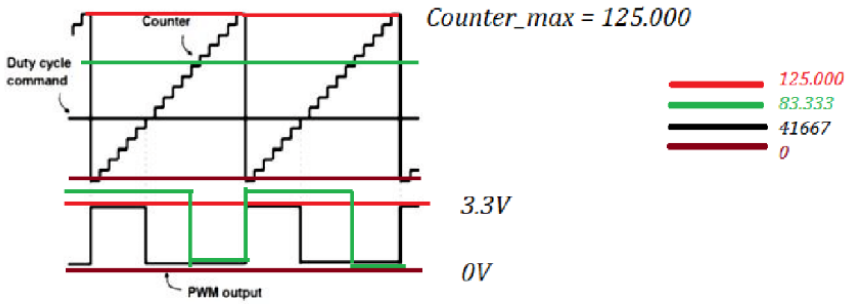

Figura 8: *PWM*

- Si se fijan hay intensidades que se repiten para los colores, por lo que en total solo tienen que cargar en el .COE [160, 32, 255, 224, 80, 208] intensidades (transformadas a su respectivo Counter_max). Para esto **investiguen (o vayan a la ayudantía)** qué hace cada BIT de los archivos *addr* y *data*, específicamente los bits de **la línea 5 de la siguiente imagen**:

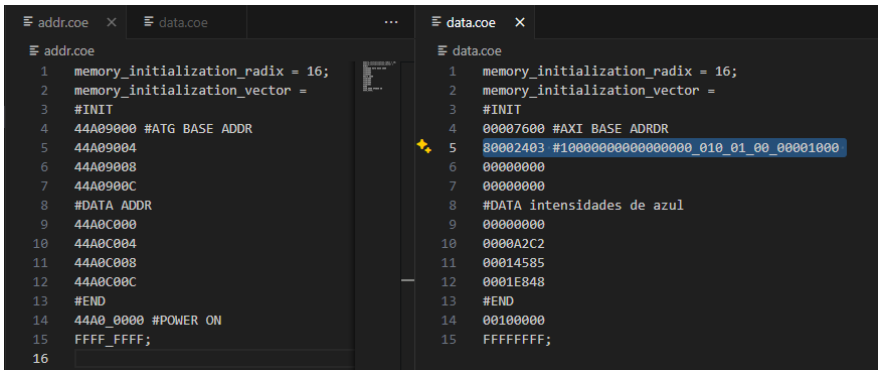

Figura 9: *COE usado en actividad previa*

- Modifique el IPCore, defina las salidas de los tres LEDs RGB para generar los colores pedidos.
- Cree un código que reciba de entrada los switches y genere el color correspondiente.
- **HINT: es posible que en el archivo de menor jerarquía les convenga investigar cómo se usan dentro del código las siguientes definiciones:**

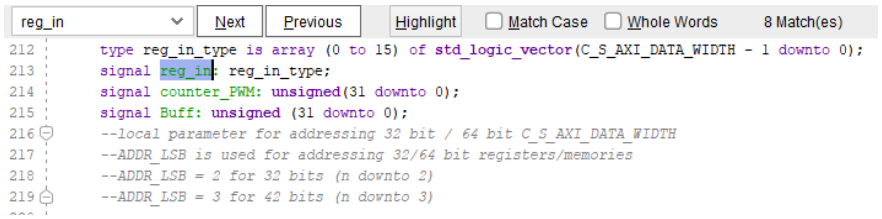

Figura 10: *Señales*