



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA
IEE2463 SISTEMA ELECTRÓNICOS PROGRAMABLES

Ayudantía 04 2S23

ZYBOZ-Entity y Architecture con VHDL

Ayudante: Agustín Kamke akamke@uc.cl
Prof. Dr.-Ing. Félix Rojas - felix.rojas@uc.cl

1. Objetivo de la Ayudantía

- Comprender el manejo de buses de datos y direcciones en un microprocesador.
- Adquirir familiaridad con la generación y lectura de datos y direcciones en un microprocesador.
- Implementar los bloques ILA (Integrated Logic Analyzer) y VIO (Virtual Input/Output) proporcionados por Vivado en un proyecto específico.

2. Actividades Previas a la Ayudantía

En el video se utiliza de base la ayudantía anterior para implementar los bloques ILA y VIO como forma de generar y leer buses datos en diversas partes del microprocesador.

- Abrir el .zip de la ayudantía anterior y ejecutar el proyecto, ya que se trabaja sobre la base de esta.
- Editar el paquete de configuración del IP Core, haciendo click derecho sobre el la unidad de control en sources y abrir edit package.
- En la sección de código de la unidad de control, realizar el cambio que lleva al sistema a reiniciar al estado inicial una vez compleado el ciclo.
- Ajustar la constante del contador en el archivo de la control unit. Esto permitirá que el contador se incremente más rápidamente para poder ser leído en ILA.

- Reemplazar los bloques debouncer y SM(State Machine) por el bloque VIO, el cual se configura como en los anexos, tal como se describe en el video, Además es necesario comentar el botón y lo switches en el .xdc una vez eliminada la SM.
- Finalmente se agrega en bloque ILA y se configura como sale en los anexos.
- El diagrama de bloques final debería verse de la siguiente manera:

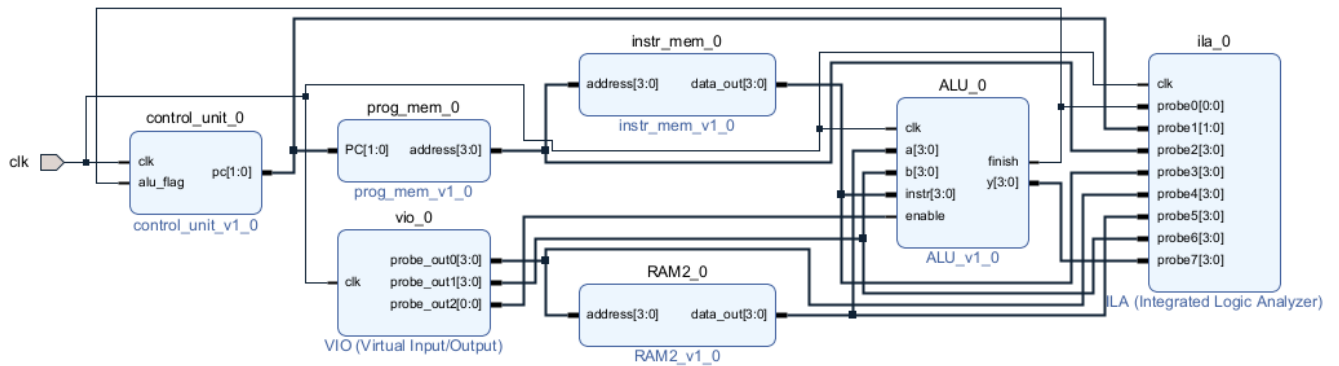


Figura 1: Esquemático final

3. Actividades Durante la Ayudantía

En nuestro esquemático actual, estamos trabajando con datos de un tamaño de 4 bits, lo que hace que nuestro sistema sea notablemente sencillo. Esto resulta ideal si deseamos aprovechar los 4 LEDs disponibles en la placa Zybo para representar información de manera clara y concisa. Sin embargo, en el mundo de los microprocesadores modernos, es común encontrar buses de datos de 32 bits, 64 bits o incluso más. Estos buses de datos más amplios permiten el procesamiento y la transferencia de información mucho más grande y compleja en una sola operación, lo que es esencial para aplicaciones de alto rendimiento y cálculos intensivos.

- Modificar los bloques RAM y ALU, en el edit package para que los datos que sea capaz de procesar ahora sean del tamaño de un byte (8 bits).
- Modificar VIO e ILA, de acuerdo a los requerimientos anteriores para manejar de manera optima este microprocesador y realizar la lectura de datos.
- (Opcional) Modificar todos los bloques del sistema de manera que el microprocesador completo, trabaje en buses de un byte, desde la control unit hasta la ALU.

Anexos

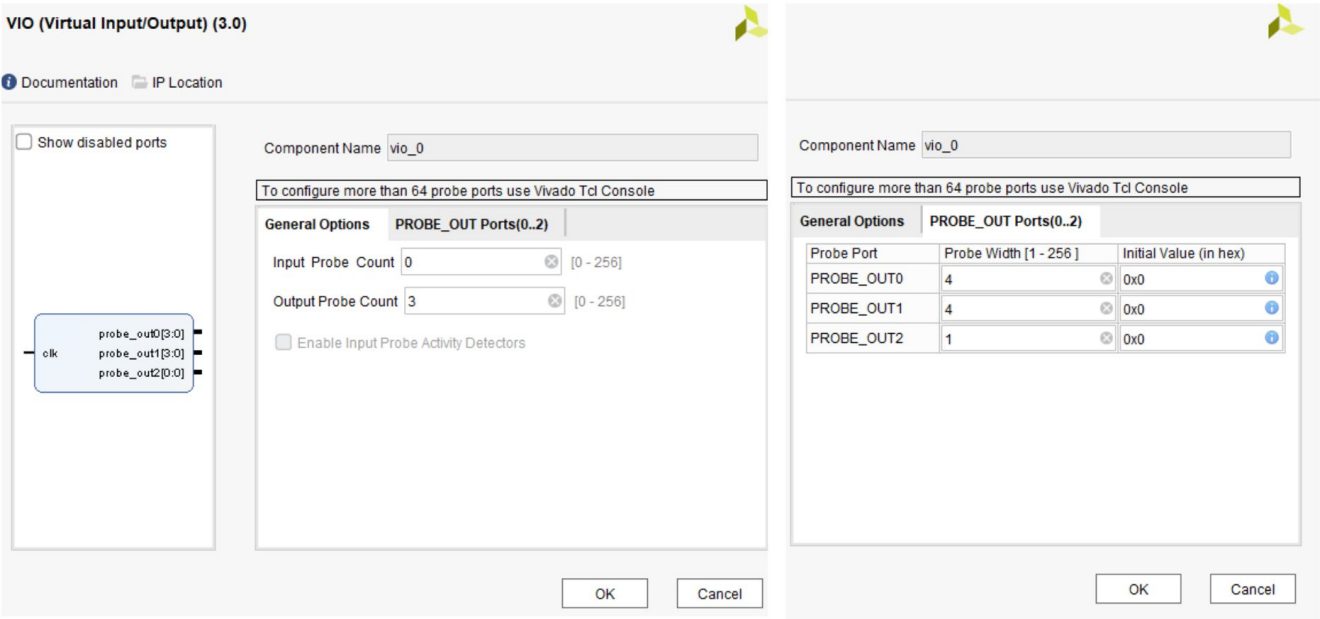


Figura 2: Configuración bloque VIO

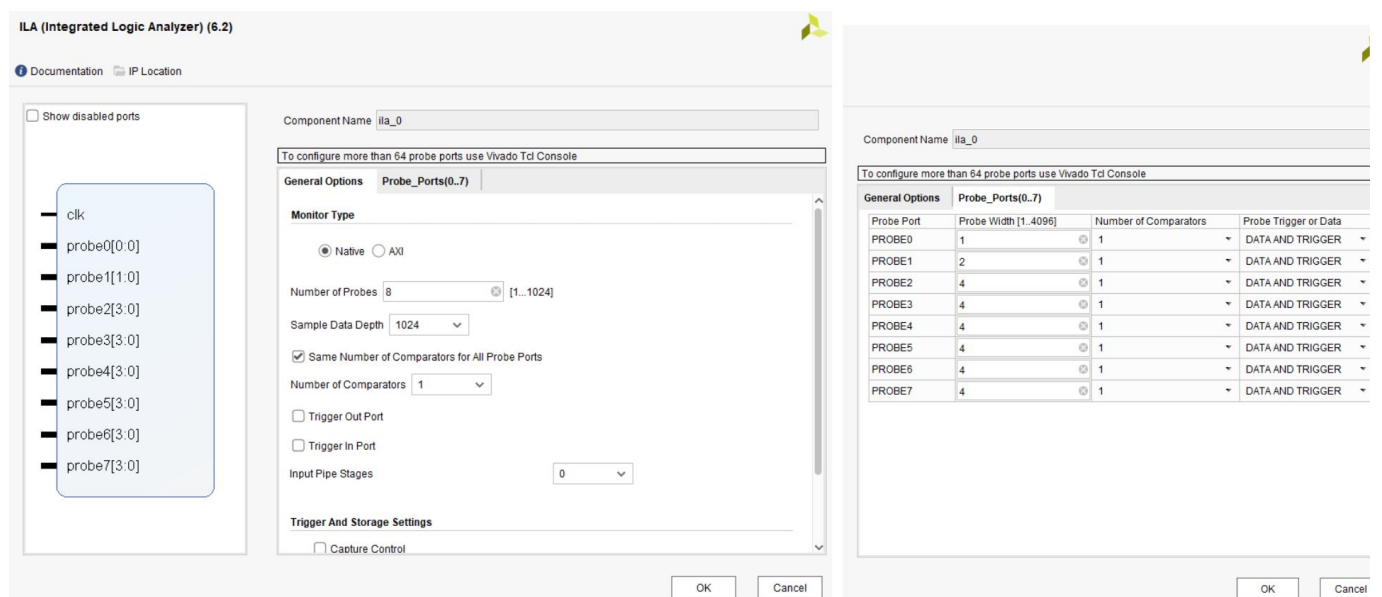


Figura 3: Configuración bloque ILA