

Circuitos lógicos programables

Práctica 2

Sintetizado de un sistema

Introducción

En esta práctica se verá el proceso de síntesis y el efecto de cambiar la configuración de la herramienta de síntesis. Se hará un análisis del sistema sintetizado y de los reportes generados.

Objetivos

Después de completar esta práctica será capaz de:

- Usar un archivo XDC provisto para aplicar restricciones de tiempo
- Elaborar el sistema y analizar las salidas generadas por las herramientas
- Sintetizar el sistema con las restricciones de tiempo provistas
- Analizar los resultados generados por las herramientas de síntesis
- Cambiar la configuración de la herramienta de síntesis y ver los efectos que produce en el sistema sintetizado
- Escribir un punto de chequeo (*checkpoint*) luego de la síntesis para que los resultados puedan ser analizados posteriormente

Procedimiento

Esta práctica está separada en pasos que consisten en sentencias generales que proveen información sobre las instrucciones detalladas que le siguen. Siga estas instrucciones detalladas para avanzar dentro de esta práctica.

Esta práctica está compuesta por 4 pasos principales: crear un proyecto usando Vivado, elaborar el sistema, sintetizar el sistema, y leer los puntos de chequeo.

Descripción del Diseño

El sistema consiste en un receptor serie asincrónico (UART) que recibe caracteres y muestra la representación binaria de la parte baja del carácter en 4 Leds. Cuando se presiona un botón, se representa la parte alta del carácter. En la Figura 1 se ve el diagrama en bloques del sistema.

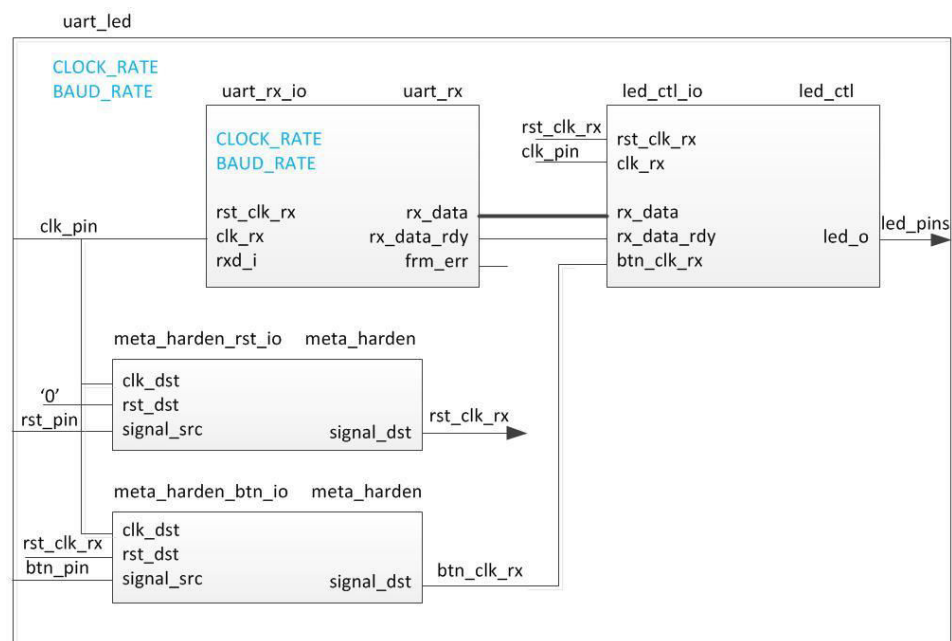
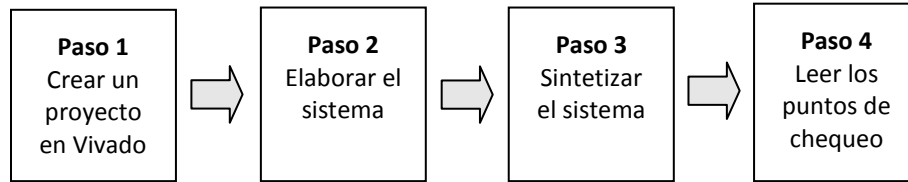


Figura 1. Sistema a implementar

Flujo General para esta práctica



Crear un Proyecto en Vivado

Paso 1

- 1-1. Iniciar la herramienta Vivado y crear un proyecto con la placa Arty Z7. Agregar los archivos HDL y las restricciones `uart_led_pins_ArtyZ7.xdc` y `uart_led_timing_ArtyZ7.xdc`.
- 1-1-1. Iniciar la herramienta Vivado
- 1-1-2. Presionar **Create Project** para iniciar el asistente. Aparecerá el cuadro de diálogo **Create A New Vivado Project**. Presionar **Next**.
- 1-1-3. Presionar el botón de navegación del campo **Project location** del formulario **New Project**, elegir la ubicación del proyecto y presionar **Select**.
- 1-1-4. Ingresar el nombre del proyecto en el campo **Project name**. Verificar que la opción **Create Project Subdirectory** esté seleccionada. Presionar **Next**.
- 1-1-5. Seleccionar la opción **RTL Project** en el formulario **Project Type**, y presionar **Next**
- 1-1-6. Verificar que esté seleccionada la opción VHDL en el menú desplegable **Target Language** y Mixed en **Simulator Language**, en el formulario **Add Sources**
- 1-1-7. Presionar el botón **Add Files** y agregar los archivos VHDL (`led_ctl.vhd`, `meta_harden.vhd`, `uart_baud_gen.vhd`, `uart_led.vhd`, `uart_rx.vhd`, `uart_rx_ctl.vhd` and `uart_top.vhd`), presionar **OK**, y después presionar **Next** para pasar al formulario **Add Constraints**

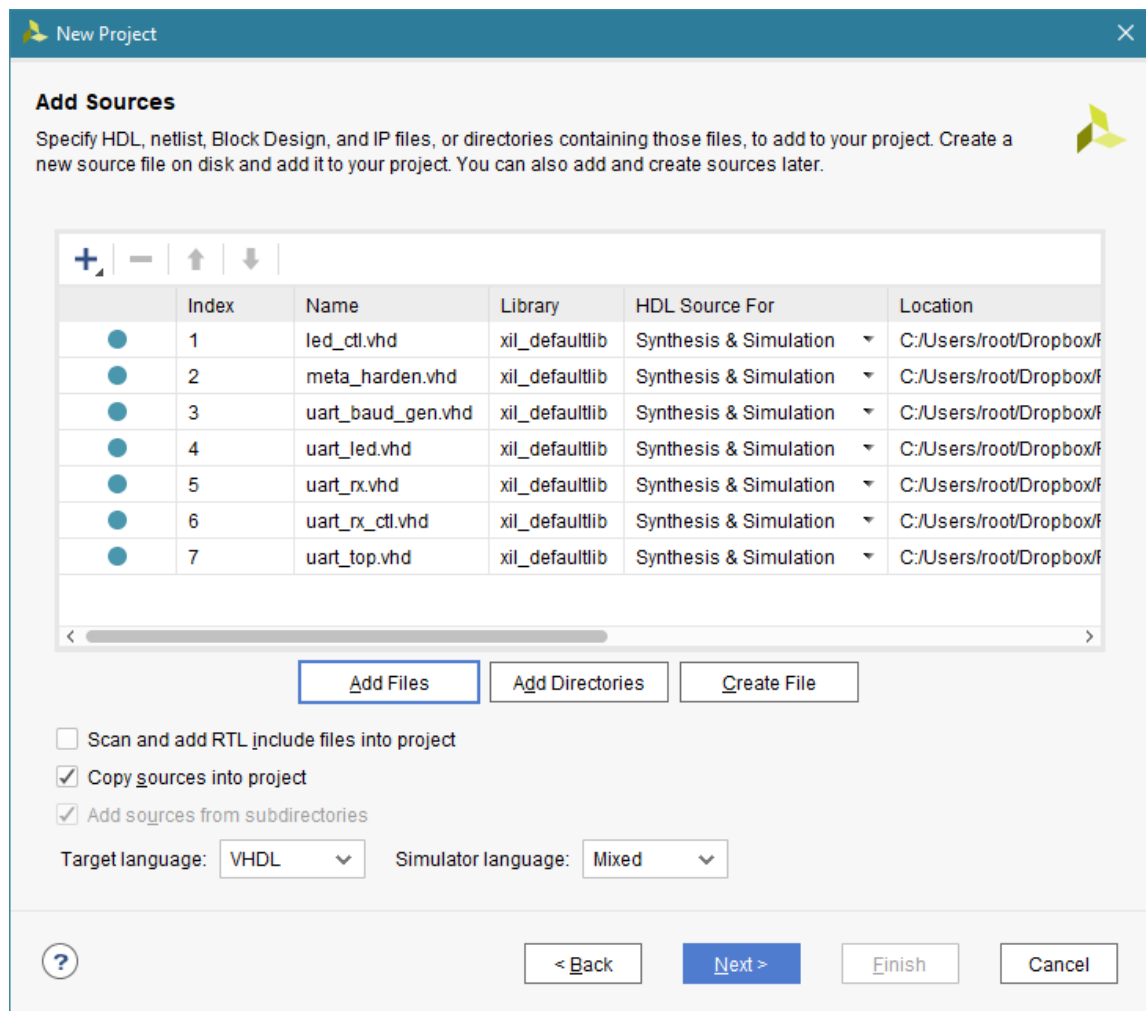


Figura 2. Agregado de los archivos fuente

- 1-1-8. Presionar el botón **Add File** y agregar el archivo de restricciones **uart_led_timing_ArtyZ7.xdc** y presionar **Ok**. Después presionar **Next**. Este archivo de restricciones asigna las restricciones de temporización (periodo, retardo de entrada y retardo de salida) del sistema.
- 1-1-9. En el formulario **Default Part**, usar la opción **Boards**. Seleccionar la placa **Arty Z7-10**. Presionar **Next**. Debería verse el resumen de configuración del proyecto.

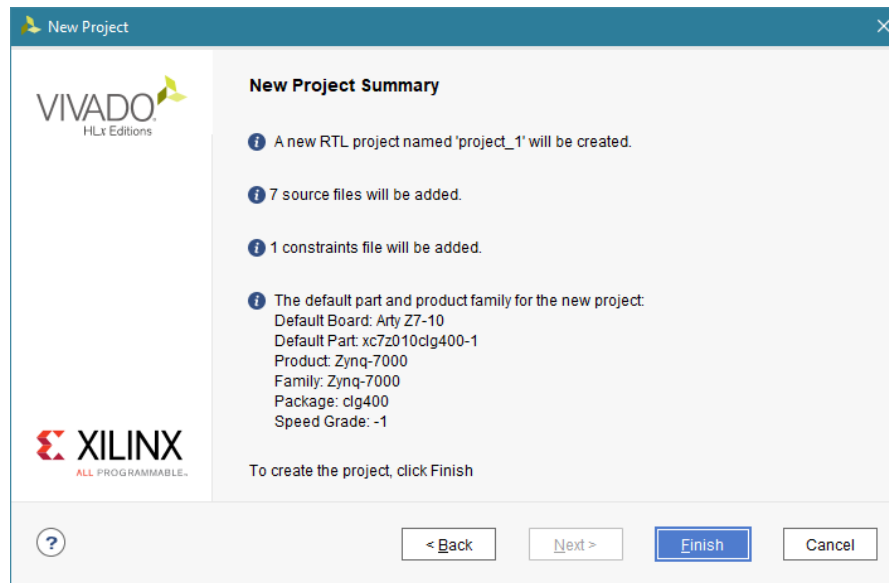


Figura 3. Resumen del proyecto

1-1-10. Presionar **Finish** para crear el proyecto.

1-2. Analizar la estructura del proyecto.

1-2-1. En el panel **Sources**, expandir `uart_top` y ver la estructura de los módulos.

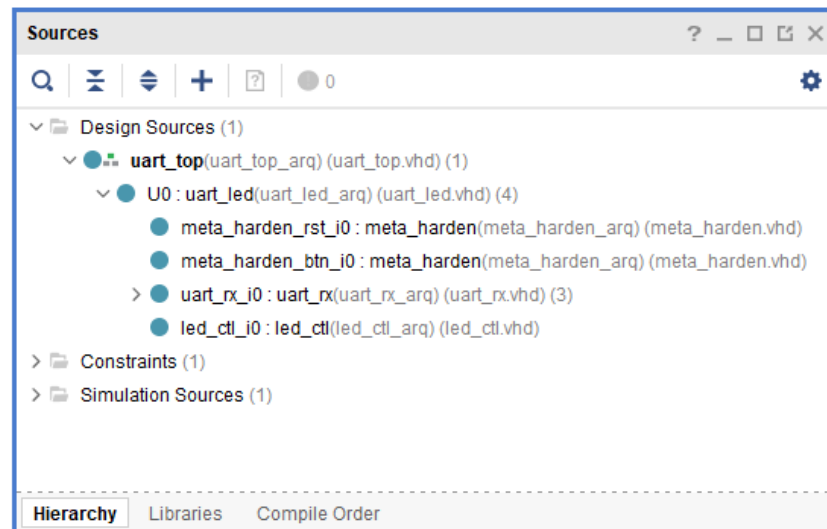


Figura 4. Estructura del proyecto

1-2-2. Presionar dos veces en `uart_led` para ver su contenido.

Ver que en el código VHDL, los parámetros `BAUD_RATE` y `CLOCK_RATE` están definidos como 115200 y 50 MHz respectivamente. Verificar que están instanciados otros módulos. Los módulos `meta_harden` se utilizan para sincronizar las entradas externas de reset y el botón para cambio de nibble.

1-2-3. Expandir las instancias U0 y uart_rx_i0 para ver su estructura

Este módulo utiliza un generador de baud rate y una máquina de estados. El pin *rx_d_pin* se muestrea a 16 veces la frecuencia del baudrate

1-3. Abrir el archivo de restricciones uart_led_timing_ArtyZ7.xdc y analizar su contenido.**1-3-1.** En el panel **Sources**, expandir la carpeta **Constraints** y presionar dos veces en el archivo uart_led_timing_ArtyZ7.xdc para abrirlo.

En la línea 4 se crea una restricción de periodo de 20 ns con un ciclo activo del 50% para la señal de reloj (clk_pin). La línea 7 crea un reloj virtual de 12 ns. Este reloj puede verse como un dispositivo externo que está generando su salida sincronizada con este reloj. Las entradas *rst_pin* y *btn_pin* están restringidas respecto a este reloj virtual (líneas 9, 10, 12 y 13). Asimismo las salidas *led_pins* también están restringidas respecto al reloj virtual.

Elaborar el sistema

Paso 2

2-1. Elaborar el sistema y realizar un análisis de la estructura RTL

- 2-1-1. En el panel **Flow Navigator**, en la sección **RTL Analysis**, expandir la opción **Open Elaborated Design** y seleccionar **Schematic**. Presionar **OK**.

El sistema será elaborado y se mostrará un diagrama esquemático del mismo.

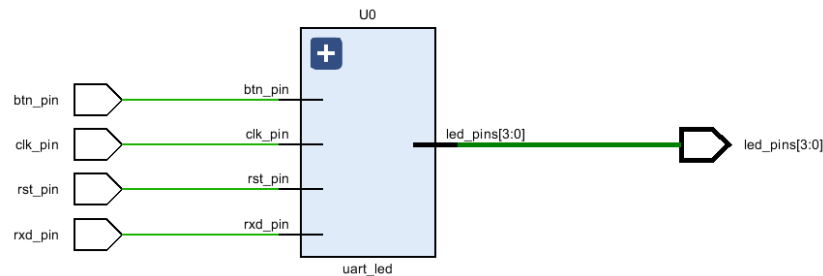


Figura 5. Diagrama esquemático

Hacer click sobre el símbolo + para desplegar el sistema

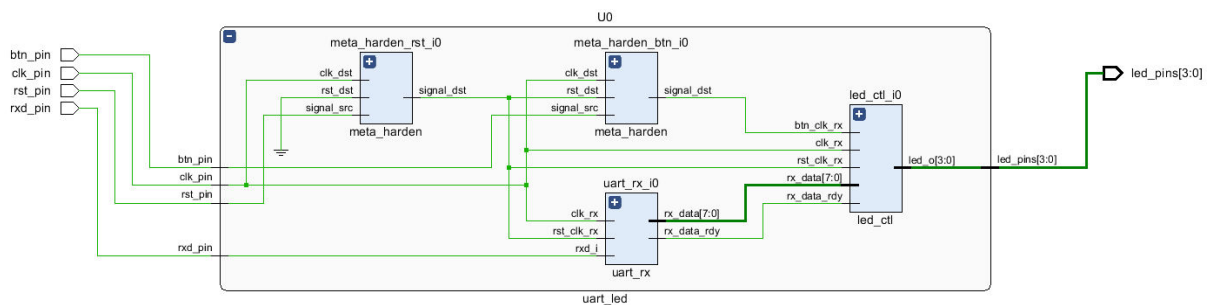


Figura 6. Diagrama lógico del sistema

Dentro del componente *U0* se ven dos instancias del componente *meta_harden*, una instancia del componente *uart_rx*, y una instancia del componente *led_ctl*.

- 2-1-2. Para ver la descripción del componente *uart_rx_i0* seleccionarlo y presionar con el botón derecho del mouse y seleccionar **Go To Source**. Se abrirá el código fuente en la sección donde se instancia el componente.
- 2-1-3. Volver al esquemático anterior y presionar dos veces en la instancia *uart_rx_i0* para ver su estructura interna.

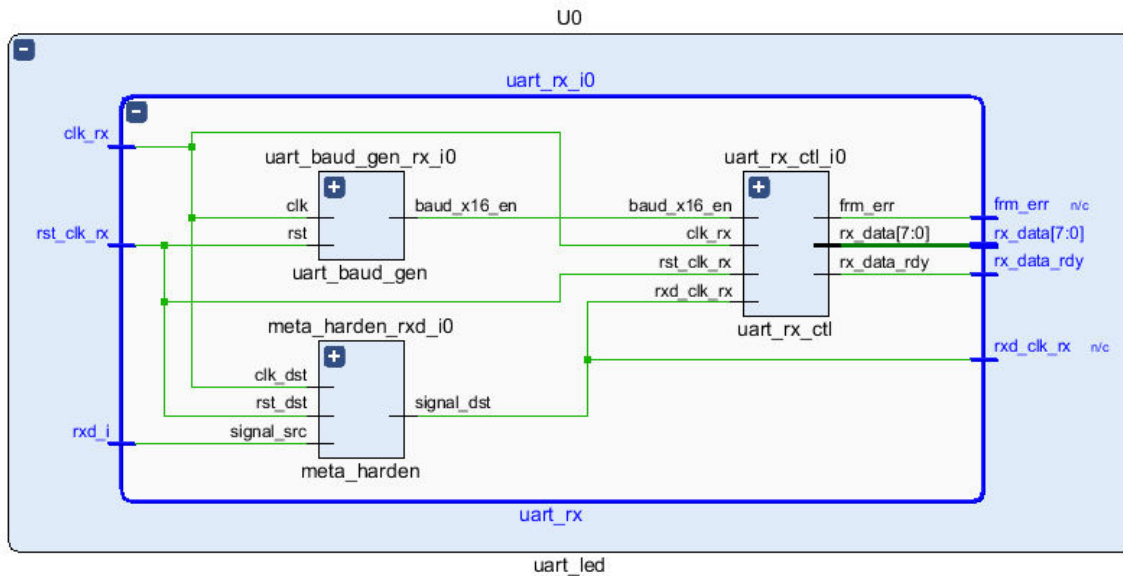


Figura 7. Estructura interna del módulo `uart_rx_i0`

- 2-1-4. En el panel **Flow Navigator**, en la sección **RTL Analysis**, expandir la opción **Open Elaborated Design** y seleccionar **Report Noise**.
- 2-1-5. Presionar **OK** para generar el reporte (de nombre `ssn_1`).
- 2-1-6. Abrir el reporte `ssn_1` y ver que hay puertos sin ubicación (unplaced ports), las secciones **Summary** y **I/O Bank Details** están resaltadas en rojo porque no se hicieron las asignaciones de pines. Sólo se marcan los pines de salida porque el análisis de ruido sólo se hace sobre pines de salida.

Tcl ConsoleMessagesLogReportsDesign RunsNoise x? _ □ ▢

SummaryMessages (2)I/O Bank DetailsLinks

</

Figura 8. Reporte de ruido

- 2-1-7. En **Project Navigator**, presionar **Add Sources**, seleccionar **Add or Create Constraints** y presionar **Next**.
- 2-1-8. Presionar el botón **Add Files** y agregar el archivo `uart_led_pins_ArtyZ7.xdc`, presionar **OK**, y luego **Finish** para agregar la restricción de ubicación de pines.

Como los archivos fuente han cambiado, la herramienta detecta esto y muestra una advertencia

ELABORATED DESIGN - xc7z010clg400-1 (active)

⚠ Elaborated Design is out-of-date. Constraints were modified. [details](#) [Reload](#)

2-1-9. Presionar el enlace **Reload**. Esto procesará las nuevas restricciones.

2-1-10. En el panel **Flow Navigator**, en la sección **RTL Analysis**, expandir la opción **Open Elaborated Design** y seleccionar **Report Noise**. Presionar **OK** para generar el reporte ssn_1. Observar que ahora no hay errores (no hay resaltados en rojo).

Tcl Console	Messages	Log	Reports	Design Runs	Noise					
Summary	Messages (1)									
I/O Bank Details										
Links										
I/O Bank Details										
Name	Port	I/O Std	Vcco	Slew	Drive Strength (...)	Off-Chip Termina...	Remaining Margin ...	Notes		
I/O Bank 0 (0)										
I/O Bank 13 (0)										
I/O Bank 34 (2)		LVC MOS33	3.30	SLOW	12	FP_VTT_50				
R14	led_pins[0]	LVC MOS33	3.30	SLOW	12	FP_VTT_50	93.04			
P14	led_pins[1]	LVC MOS33	3.30	SLOW	12	FP_VTT_50	94.90			
I/O Bank 35 (2)		LVC MOS33	3.30	SLOW	12	FP_VTT_50				
N16	led_pins[2]	LVC MOS33	3.30	SLOW	12	FP_VTT_50	93.36			
M14	led_pins[3]	LVC MOS33	3.30	SLOW	12	FP_VTT_50	90.69			

Figura 9. Reporte de ruido sin errores

Sintetizar el sistema

Paso 3

3-1. Sintetizar el sistema con la herramienta de síntesis y analizar los resultados.

- 3-1-1.** En el menú **Flow Navigator**, en la sección **Synthesis**, seleccionar **Run Synthesis**. Presionar **OK**.

El proceso de síntesis se ejecutará sobre el componente `uart_top.vhd` y todos sus subcomponentes. Cuando el proceso finalice, aparecerá el cuadro de diálogo **Synthesis Completed** con 3 opciones.

- 3-1-2.** Seleccionar la opción **Open Synthesized Design** y presionar **OK** para ver los resultados del proceso de síntesis.

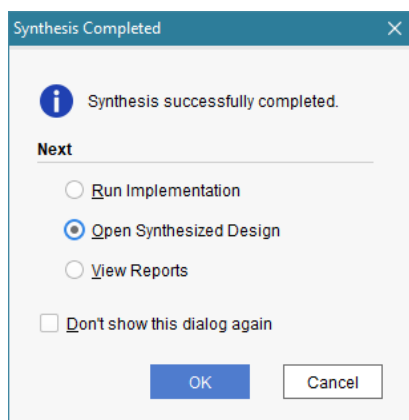


Figura 10. Selección para abrir el diseño sintetizado

Presionar **Yes** para cerrar el sistema elaborado si aparece el cuadro de diálogo correspondiente.

- 3-1-3.** Seleccionar la pestaña **Project Summary**.

Si no está visible, seleccionar Layout ► Default Layout. En la pestaña **Project Summary**, en la sección **Utilization** (al final del reporte) presionar el enlace **Table** y tomar nota de la información solicitada (utilización estimada de recursos):

LUT	_____
FF	_____
IO	_____
BUFG	_____

- 3-1-4.** En el menú **Flow Navigator**, seleccionar **Synthesis**, seleccionar **Open Synthesized Design** y presionar **Schematic**.

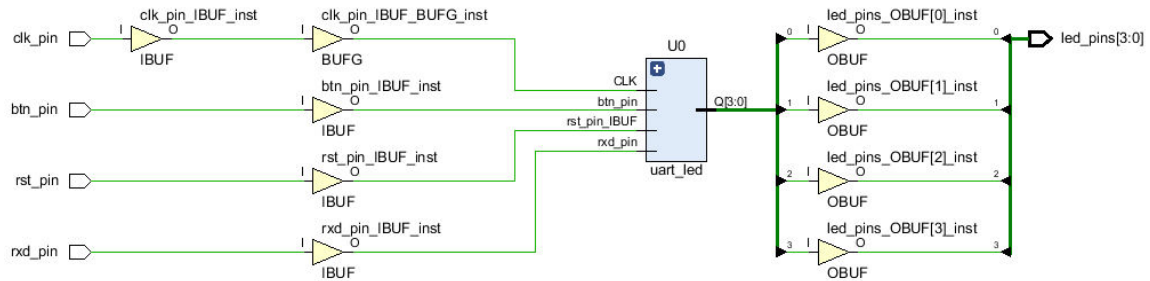


Figura 11. Diagrama esquemático del sistema sintetizado

Los bloques IBUF y OBUF se instancian automáticamente y se agregan al sistema para las señales de entradas y salidas.

- 3-1-5.** Presionar dos veces en la instancia U0 y luego en la instancia uart_rx_i0 para ver su estructura interna.

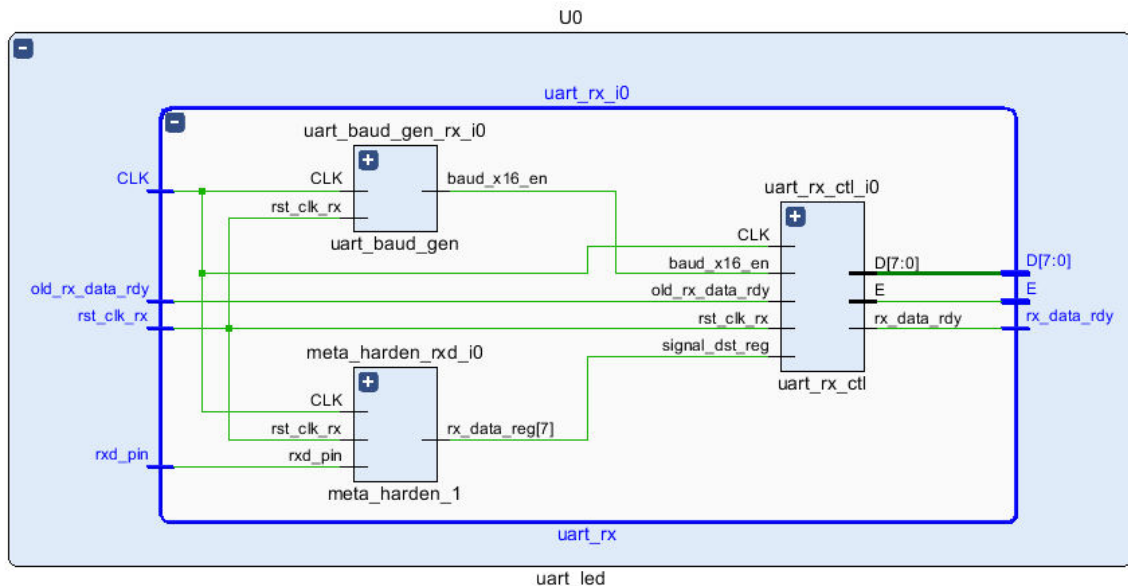


Figura 12. Estructura interna de la instancia uart_rx_i0

- 3-1-6.** Seleccionar la instancia uart_baud_gen_rx_i0, presionar el botón derecho y seleccionar **Go To Source**.

Se abrirá el archivo fuente correspondiente posicionado en una línea en particular, donde está instanciado el componente. A su vez, se puede observar cómo son pasados por parámetro los valores CLOCK_RATE y BAUD_RATE.

- 3-1-7.** Volver al diagrama esquemático y presionar dos veces en la instancia meta_harden_rxd_io para ver como se implementa el circuito de sincronización mediante dos FFs. Este circuito es necesario para reducir la probabilidad de metaestabilidad.

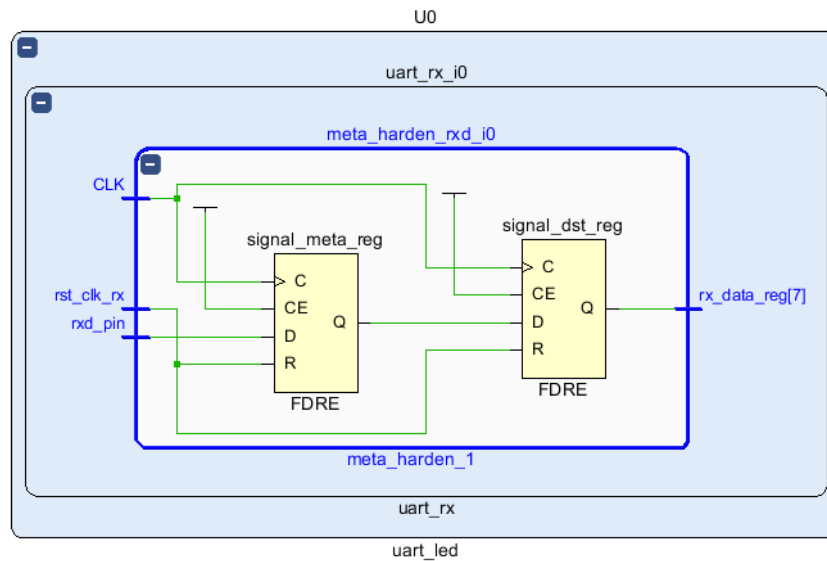



Figura 13. Circuito de sincronización

3-1-8. Presionar el botón  en la vista de diagrama esquemático para subir de nivel en la estructura del sistema.

3-2. Análisis del reporte de temporización.

3-2-1. En el menú **Flow Navigator**, seleccionar **Synthesis**, luego **Open Synthesized Design** y finalmente presionar **Report Timing Summary**

3-2-2. Presionar **OK** para generar el reporte timing_1 (nombre por defecto).

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): -3.970 ns	Worst Hold Slack (WHS): -1.508 ns	Worst Pulse Width Slack (WPWS): 9.500 ns
Total Negative Slack (TNS): -15.790 ns	Total Hold Slack (THS): -2.983 ns	Total Pulse Width Negative Slack (TPWS): 0.000 ns
Number of Failing Endpoints: 4	Number of Failing Endpoints: 2	Number of Failing Endpoints: 0
Total Number of Endpoints: 103	Total Number of Endpoints: 103	Total Number of Endpoints: 46

Timing constraints are not met.

Figura 14. Reporte de temporización Arty Z7

Notar que las entradas **Design Timing Summary** e **Inter-Clock Paths** del panel de la izquierda están resaltadas en rojo, indicando violaciones de temporización. En el panel de la derecha la información está dividida en tiempo de establecimiento, retención y ancho de pulso (Setup, Hold, y Pulse Width).

En la columna **Setup** existe un link (**Worst Negative Slack (WNS)**) que siguiéndolo permite ver cómo se arma el camino de temporización. El indicador **Total Negative Slack (TNS)** está resaltado en rojo indicando la cantidad total de tiempo de las violaciones de temporización del

sistema y el **Number of Failing Endpoints** indica la cantidad total de caminos de temporización que no cumplen los requisitos de temporización.

3-2-3. Presionar el enlace de **WNS** y verificar los 4 caminos que no cumplen la temporización.

Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement
Path 21	-3.970	1	1	1	U0/led_ctl_i0/le_ne_reg[1]/C	led_pins[1]	5.003	4.203	0.800	4.0
Path 22	-3.965	1	1	1	U0/led_ctl_i0/le_ne_reg[0]/C	led_pins[0]	4.998	4.198	0.800	4.0
Path 23	-3.929	1	1	1	U0/led_ctl_i0/le_ne_reg[3]/C	led_pins[3]	4.962	4.162	0.800	4.0
Path 24	-3.927	1	1	1	U0/led_ctl_i0/le_ne_reg[2]/C	led_pins[2]	4.960	4.160	0.800	4.0

Figura 15. Caminos de temporización que no cumplen los requisitos de temporización

3-2-4. Presionar 2 veces en Path 21 para ver como se arma ese camino

Summary				
Name	Path 21			
Slack	-3.970ns			
Source	U0/led_ctl_i0/led_pipeline_reg_reg[1]/C (rising edge-triggered cell FDRE clocked by clk_pin (rise@0.000ns fall@10.000ns period=20.000ns))			
Destination	led_pins[1] (output port clocked by virtual_clock (rise@0.000ns fall@6.000ns period=12.000ns))			
Path Group	virtual_clock			
Path Type	Max at Slow Process Corner			
Requirement	4.000ns (virtual_clock rise@24.000ns - clk_pin rise@20.000ns)			
Data Path Delay	5.003ns (logic 4.203ns (84.015%) route 0.800ns (15.985%))			
Logic Levels	1 (OBUF=1)			
Output Delay	0.000ns			
Clock Path Skew	-2.942ns			
Clock Uncertainty	0.025ns			
Clock Dom...Crossing	Inter clock paths are considered valid unless explicitly excluded by timing constraints such as set_clock_groups or set_false_path.			

Source Clock Path				
Delay Type	Incr (ns)	Path (...)	Locati...	Netlist Resource(s)
(clock clk_pin rise edge)	(r) 20.000	20.000		
	(r) 0.000	20.000	Sit..16	clk_pin
net (fo=0)	0.000	20.000	Sit..16	clk_pin
			Sit..16	clk_pin_IBUF_inst/I
IBUF (Prop ibuf I O)	(r) 1.457	21.457	Sit..16	clk_pin_IBUF_inst/O
net (fo=1, unplaced)	0.800	22.257		clk_pin_IBUF
				clk_pin_IBUF_BUFG_inst/I
BUFG (Prop bufg I O)	(r) 0.101	22.358		clk_pin_IBUF_BUFG_inst/O
net (fo=45, unplaced)	0.584	22.942		U0/led_ctl_i0/CLK
FDRE				U0/led_ctl_i0/led_pipeline_reg_reg[1]/C

Figura 16. Detalle de un camino de temporización que no cumple los requisitos

3-3. Generación de los reportes de utilización y consumo.

3-3-1. Presionar **Report Utilization** dentro de **Open Synthesized Design**, y presionar **OK** para generar el reporte de utilización. Presionar **Summary** en el panel de la izquierda.

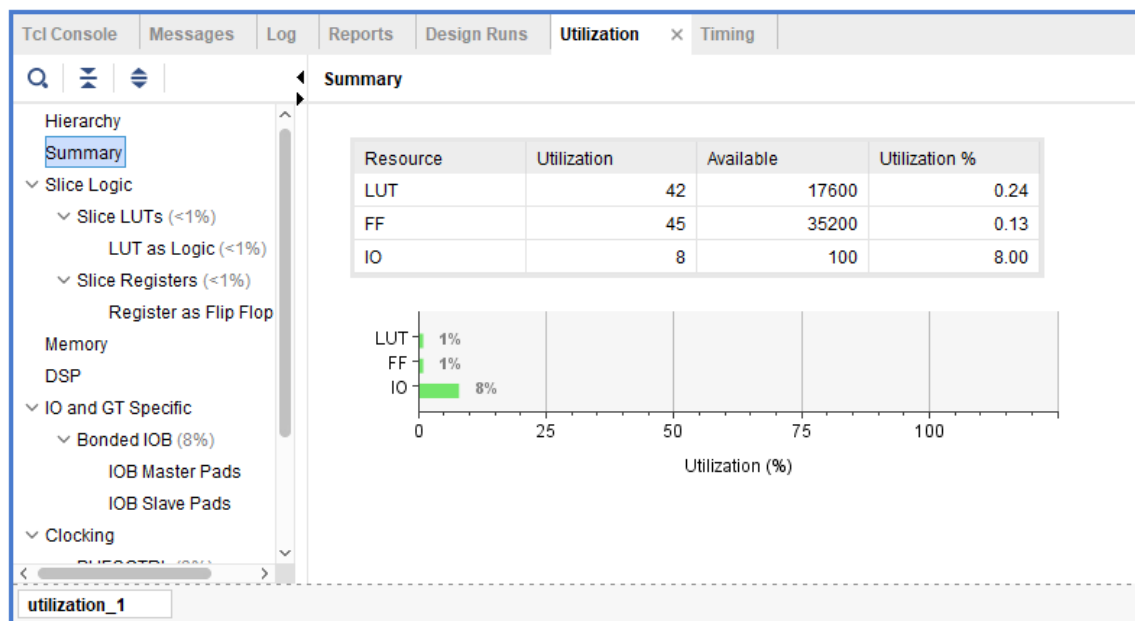


Figura 17. Resumen del reporte de utilización

- 3-3-2.** Seleccionar la entrada **Slice Logic** en el panel de la izquierda y ver la utilización de cada instancia (Slice Logic ► Slice LUTs ► LUT as Logic). Expandir las instancias del panel de la derecha para ver la utilización de cada subcomponente.

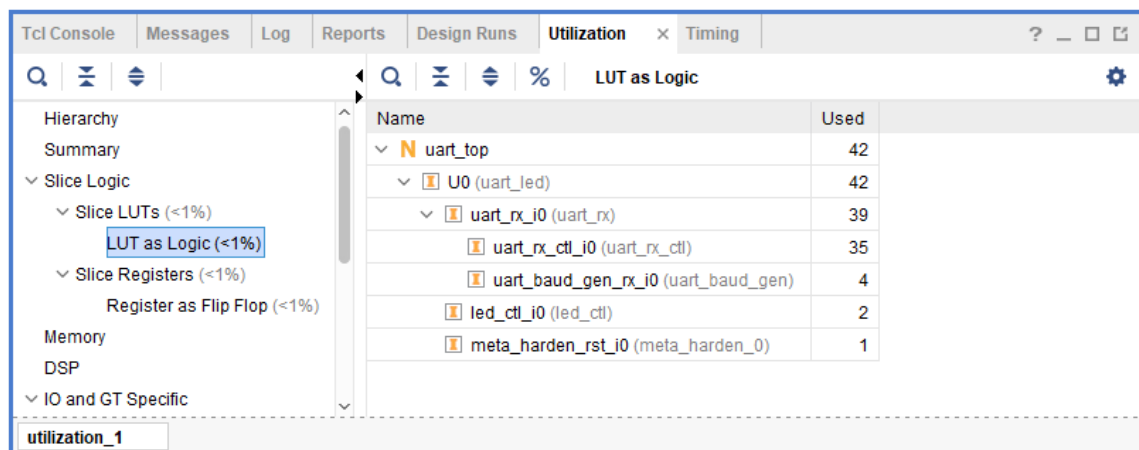


Figura 18. Utilización de recursos por cada modulo

- 3-3-3.** Presionar **Report Power** en **Open Synthesized Design**, y presionar **OK** para generar la estimación de consumo del sistema.

En este punto el reporte es una estimación, ya que no hay información adicional sobre la utilización del sistema.

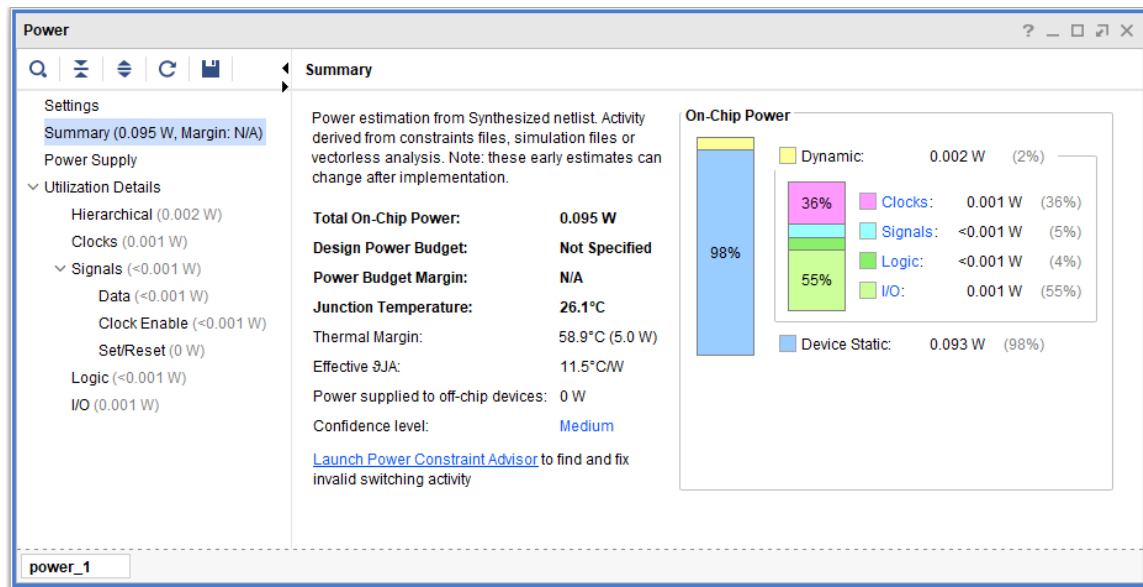


Figura 19. Estimación de consumo

3-4. Escribir el punto de chequeo para analizar los resultados posteriormente sin necesidad de volver a sintetizar el sistema.

- 3-4-1. Seleccionar el menú File ► Checkpoint ► Write... para grabar el sistema en el estado actual, de manera que pueda ser abierto posteriormente para su análisis.
- 3-4-2. Aparecerá un cuadro de diálogo mostrando el nombre por defecto del archivo de punto de chequeo en el directorio del proyecto

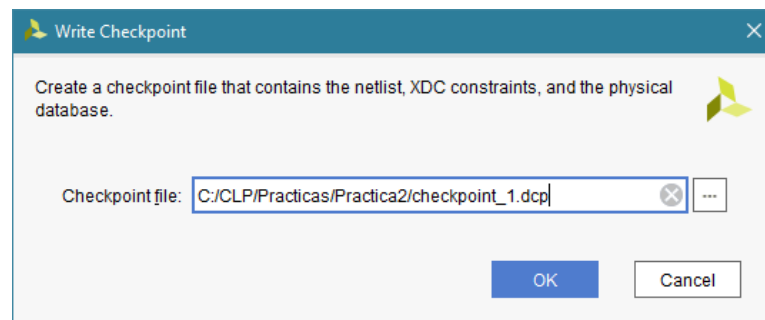


Figura 20. Cuadro de diálogo para guardar un punto de chequeo

- 3-4-3. Presionar **OK**.

3-5. Cambiar la configuración de la herramienta de síntesis para aplanar la estructura del sistema. Sintetizar nuevamente y analizar los resultados.

- 3-5-1. En **Project Manager** presionar sobre **Settings**. Se abrirá el cuadro de diálogo con la configuración del proyecto. En el panel de la izquierda, bajo **Project Settings** seleccionar **Synthesis**.

3-5-2. En la configuración **flatten_hierarchy** seleccionar full para aplanar la estructura del sistema.

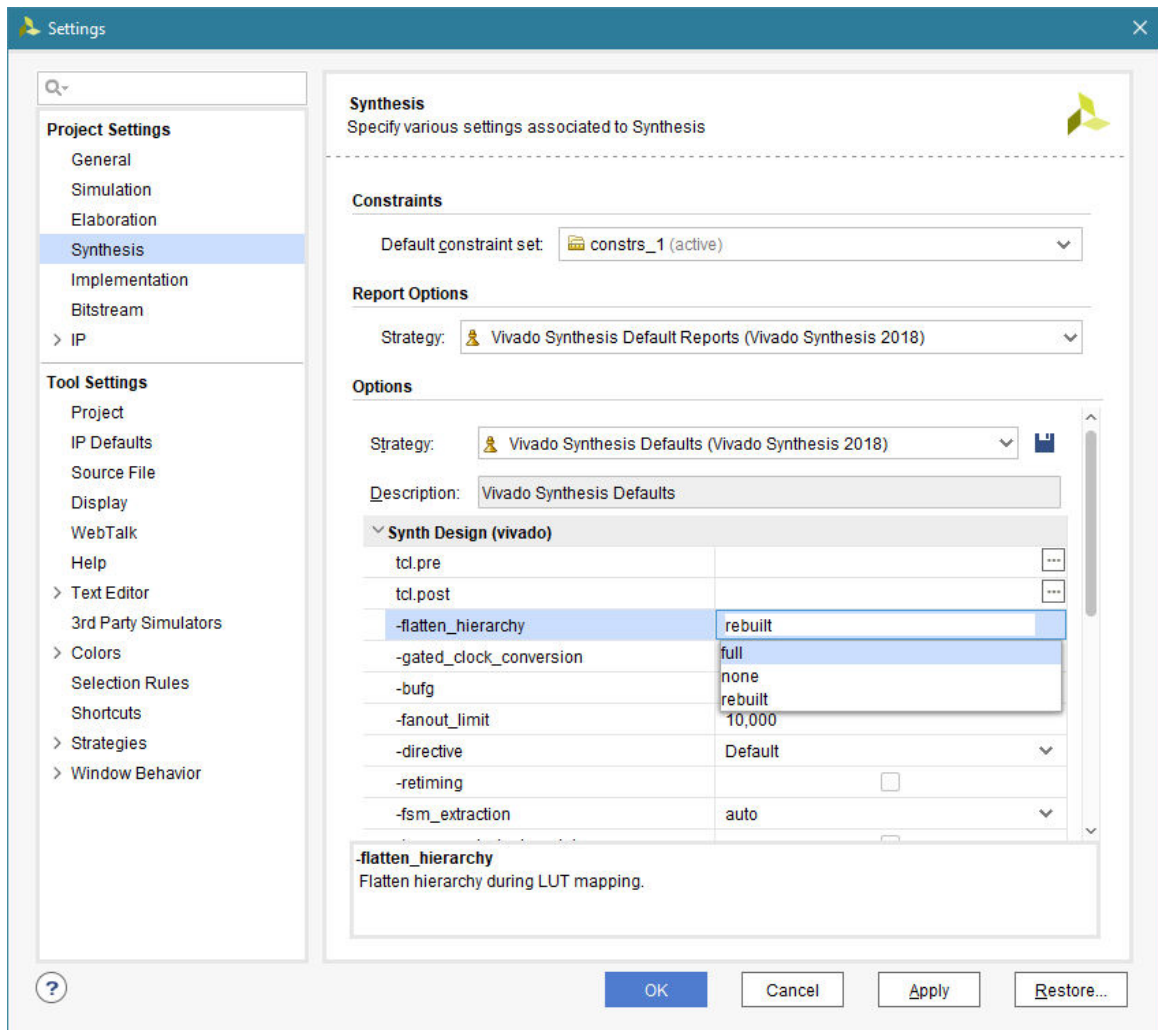


Figura 21. Opción **flatten_hierarchy**

3-5-3. Presionar **OK**.

3-5-4. Aparecerá un cuadro de diálogo **Create New Run** consultando si se debe crear una nueva síntesis independiente, ya que la configuración del proyecto cambió.

3-5-5. Presionar **Yes**.

3-5-6. Cambiar el nombre de la nueva síntesis de synth_2 a synth_flatten y presionar **OK**.

3-5-7. En el menú **Flow Navigator**, seleccionar **Synthesis**, presionar **Run Synthesis**.

3-5-8. Presionar **OK**. El sistema será sintetizado nuevamente.

3-5-9. Una vez completado el proceso de síntesis aparecerá el cuadro de diálogo **Synthesis Completed**. Seleccionar la opción **Open Synthesized Design** y presionar **OK** para abrir el sistema sintetizado.

3-5-10. En el menú **Flow Navigator**, seleccionar **Synthesis**, seleccionar **Open Synthesized Design** y presionar **Schematic** para ver el diagrama esquemático del sistema.

El sistema está totalmente aplanado y no hay una estructura de módulos y submódulos

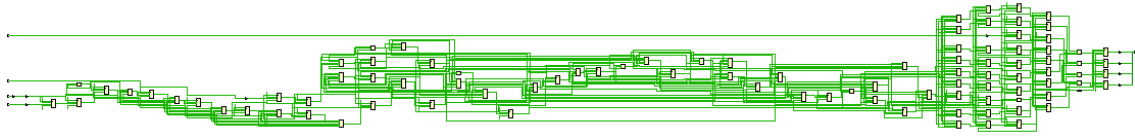


Figura 22. Sistema aplanado

3-5-11. Presionar en el reporte **Report Utilization**, presionar **OK** para generar el reporte y verificar que no está disponible la estructura jerárquica, sino que hay un único módulo. Registrar la cantidad de recursos utilizados

LUT as Logic: _____

Register as Flip Flop: _____

3-6. Guardar el nuevo punto de chequeo para analizar los resultados sin necesidad de volver a sintetizar el sistema.

3-6-1. Seleccionar el menú **File ► Checkpoint ► Write...** para grabar el sistema en el estado actual, de manera que pueda ser abierto posteriormente para su análisis.

3-6-2. Aparecerá un cuadro de diálogo mostrando el nombre por defecto del archivo de punto de chequeo en el directorio del proyecto.

3-6-3. Presionar **OK**.

3-6-4. Cerrar el Proyecto mediante el menú **File ► Close Project**. Presionar **OK**.

Leer los puntos de chequeo

Paso 4

4-1. Leer el punto de chequeo guardado (checkpoint_1) para analizar los resultados sin sintetizar nuevamente el sistema.

4-1-1. En la pantalla de inicio **Quick Start**, seleccionar File ► Checkpoint ► Open

4-1-2. Navegar hasta el directorio del proyecto y seleccionar el punto de chequeo checkpoint_1

4-1-3. Presionar **OK**.

4-1-4. Si el diagrama esquemático no se abre inicialmente, en la pestaña de redes (**Netlist**) del panel de la izquierda, seleccionar el componente U0 (uart_led), presionar el botón derecho y seleccionar **Schematic**

En la pestaña **Schematic** se verán los componentes del sistema. Presionando dos veces en un cualquiera de ellos, se accede a su estructura interna. También se puede seleccionar un subcomponente en el panel de la izquierda, presionar el botón derecho y seleccionar la opción **Schematic** para abrir su correspondiente diagrama esquemático.

4-1-5. En la pestaña **netlist**, seleccionar la instancia U0 (uart_led), presionar el botón derecho y seleccionar **Show Hierarchy**.

Se puede ver así una representación de la relación entre los distintos componentes.

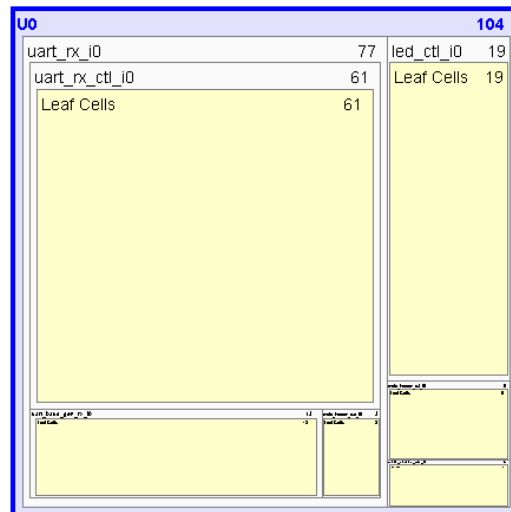


Figura 23. Jerarquía del diseño

4-1-6. Seleccionar Reports ► Timing ► Report Timing Summary y presionar **OK** para ver el correspondiente reporte.

4-1-7. Seleccionar Reports ► Timing ► Report Utilization y presionar **OK** para ver el correspondiente reporte.

4-1-8. Seleccionar File ► Checkpoint ► Open. Navegar hasta el directorio del proyecto y seleccionar el punto de chequeo checkpoint_2. Presionar **OK** para abrirlo

- 4-1-9.** En el cuadro de diálogo, presionar **No** para mantener abierto el punto de chequeo Checkpoint_1. Esto abrirá una segunda instancia del entorno gráfico.
- 4-1-10.** Si el diagrama esquemático no se abre inicialmente, en la pestaña de redes (netlist) del panel de la izquierda, seleccionar el componente uart_top, presionar el botón derecho y seleccionar **Schematic**.
En la pestaña **Schematic** se verá el sistema aplanado.
- 4-1-11.** En ambos puntos de chequeo se puede acceder a los reportes necesarios.
- 4-1-12.** Cerrar la aplicación seleccionando File ► Exit y presionando **OK** en ambos cuadros de diálogo.