Circuitos Lógicos Programables  
Trabajo Práctico final

****

**Carrera de Especialización en Sistemas Embebidos**

**Año 2024**

**Docente:**

**Ing. Nicolás Álvarez**

**Alumno**

**Ing. Quarin, Lautaro**

### 

## Introducción.

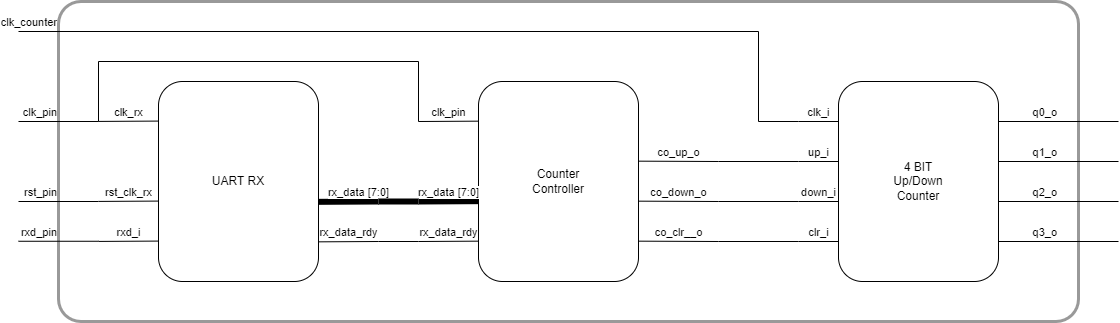
Este proyecto tiene como objetivo desarrollar un sistema digital que funcione como un contador ascendente/descendente, el cual se controla a través de la interfaz UART (Universal Asynchronous Receiver-Transmitter) y un reloj externo. El contador puede aumentar o disminuir su valor basado en los comandos recibidos a través de la UART, con el reloj externo responsable de dictar el ritmo del conteo según la configuración proporcionada por el controlador UART.

El diseño se implementó utilizando el lenguaje VHDL (Very High Speed Integrated Circuit Hardware Description Language), un lenguaje de descripción de hardware ampliamente utilizado para modelar y simular circuitos digitales. A lo largo del proyecto, se realizaron simulaciones para verificar el correcto funcionamiento del diseño, y se documentó todo el proceso, incluyendo explicaciones detalladas, diagramas de bloques, capturas de simulaciones y resultados de la implementación en la FPGA.

Finalmente, el contador desarrollado se puede controlar a través de la interfaz serial UART, lo que permite su aplicación en diversas tareas de conteo.

## Descripción

### Diagrama general



### Descripción

#### Módulo UART RX

El módulo UART RX es responsable de recibir datos en serie con la siguiente configuración

* Baud Rate: 115200
* Bits de datos: 8
* Stop bit: 1
* Parity: 1

Si el dato recibido es válido de acuerdo con la configuración del dispositivo, este dato se acepta y se convierte en un vector std\_logic\_vector que se envía al "Counter Controller" junto con la señal de "data ready".

#### Counter Controller

El módulo Counter Controller recibe el dato de la UART y lo compara con una serie de palabras predefinidas que representan comandos válidos. Si el dato coincide con alguno de estos comandos, el Counter Controller cambia el estado de la señal de salida correspondiente. Los comandos válidos incluyen:

* ‘U’ (up): Incrementar.
* ‘D’ (down): Decrementar.
* ‘C’ (clear): Reestablecre.

En caso de recibir simultáneamente las señales de UP y DOWN activadas (1), el contador dará prioridad a UP. Si ambas señales están desactivadas (0), el contador dejará de contar hasta que una de estas señales vuelva a estar en 1.

#### bit Up/Down Counter

El módulo 4-bit Up/Down Counter es el contador principal del sistema. Este módulo cuenta con un reloj independiente que regula el conteo. Además, recibe las señales de control generadas por el Counter Controller, las cuales determinan si el contador debe incrementar, decrementar o restablecer su valor, según los comandos recibidos a través de la UART.

### Analisis RTL

Utilizando la herramienta RTL análisis de vivado se llega al siguiente resultado:

UART\_CONT:

Gráfico, Gráfico en cascada

Descripción generada automáticamente  
UART\_RX

Imagen que contiene Gráfico

Descripción generada automáticamente

Imagen que contiene Gráfico

Descripción generada automáticamenteUP/DOWN\_TOP:

## Simulación

Para verificar el correcto funcionamiento del diseño completo, se desarrollaron bancos de prueba ("test benches") individuales para cada bloque y se realizó una simulación final del sistema completo. A continuación, se presentan los resultados más relevantes de la simulación del sistema completo.

### 3.1 Configuración

* Clocks:
  + ‘clk\_pin’: Reloj general para todos los bloques (excepto contador) con período de 8 nanosegundos.
  + ‘clk\_i’: Reloj específico para el contador con un período de 20 Clocks:
* Entradas:
  + ‘rxd\_pin’: Entrada de datos en serie.
* Salidas:
  + ‘cont\_pins’: Pines de salida del contador.
* Señales internas:
  + ‘co\_up\_o’: controla el incremento del contador.
  + ‘co\_down\_o’: controla el decremento del contador.
  + ‘co\_clr\_o’: Restablece el contador a 0.

### 3.2 Resultados

La simulación del diseño completo demostró el correcto funcionamiento de todos los bloques individuales y su interacción dentro de la estructura jerárquica. Los datos recibidos en Rxd\_pin son procesados por el controlador, generando las señales Co\_up\_o, Co\_down\_o y Co\_clr\_o que controlan el contador. El contador actualiza su valor en función de estas señales y envía el resultado a los pines Cont\_pins.

Interfaz de usuario gráfica

Descripción generada automáticamente

Observando de cerca podemos obtener una mejor apreciación de los marcadores:  
En la siguiente figura se observa el comportamiento inicial del sistema. Se envía la letra "U" a través de la UART, que es interpretada correctamente. El controlador genera la señal Co\_up\_o, activando el contador para que comience a contar en sentido ascendente.

Interfaz de usuario gráfica

Descripción generada automáticamente

Se envía un dato no válido ("I") a través de la UART. El controlador descarta este dato, y no se produce ninguna modificación en el comportamiento del sistema.

Interfaz de usuario gráfica

Descripción generada automáticamente

Se envía la letra "D" a través de la UART. El controlador ajusta las señales de salida, activando Co\_down\_o y desactivando Co\_up\_o, lo que provoca que el contador comience a decrementar.

Interfaz de usuario gráfica

Descripción generada automáticamente

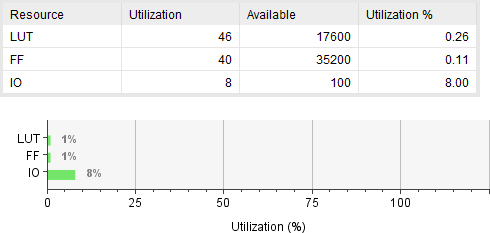
Se envía la letra "C" (clear) a través de la UART. El controlador genera un pulso en la señal Co\_clr\_o, reseteando el contador a 0 y deteniendo el conteo.

Interfaz de usuario gráfica

Descripción generada automáticamente

## Recursos

Se muestra a continuación la utilización de los recursos de la FPGA para el proyecyto



Se ve que la utilización de LUTs, FFs e IO es minima en comparación con la capacidad total de la FPGA.

Tabla

Descripción generada automáticamente

Se aprecia que el modulo uart\_rx de recepción de la UART es el que mas recursos consume, lo que era esperado ya que es el que recibe y procesa los datos. Se entiende entonces que es un diseño aceptable para un contador por UART ya que es eficiente en términos de utilización de recursos y permite que sea adaptado a proyectos mas grandes sin muchas complicaciones.

## Conclusión

Este proyecto fue una excelente oportunidad para aplicar conceptos de diseño de hardware digital y tecnologias FPGA. Aunque se trate de un proyecto relativamente sencillo, al ser el primer contacto con un proyecto de este tipo me sirvió para comprender el “lenguaje” y la forma de trabajo que requiere este tipo de tecnología.