Universitat Politècnica de Catalunya

Arquitectura de computadors d’altes prestacions

Práctica 1 - Grupo 5

Sumador de 1 bit

Alumnes:

Manuel Velastegui

Carlos Andres Rodríguez Torres

Quatrimestre primavera 2024-2025



**ÍNDICE**

[**EJERCICIO 3**](#)

[**PREGUNTAS 3**](#_u8pjsyn2rt5w)

[1. Elaboración RTL 3](#_gs7kwifkukw2)

[2. Creación fuente de onda 4](#_myr3er4mf7tu)

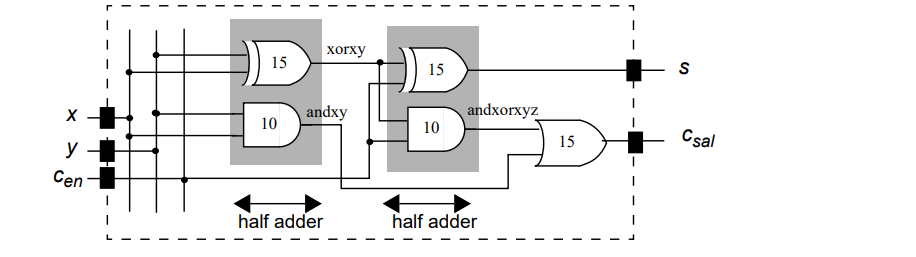
[3. Creación fuente de onda 5](#_2ucl39vj4u0j)

[Caso A 6](#_ncc8jur10yg4)

[Caso B 6](#_tp9oq28s2eea)

# **EJERCICIO**

Edite un fichero VHDL donde se especifique, utilizando un modelo de flujo de datos el sumador de 1 bit que se muestra seguidamente. Los valores numéricos en el interior de las puertas especifican el retardo en ns.

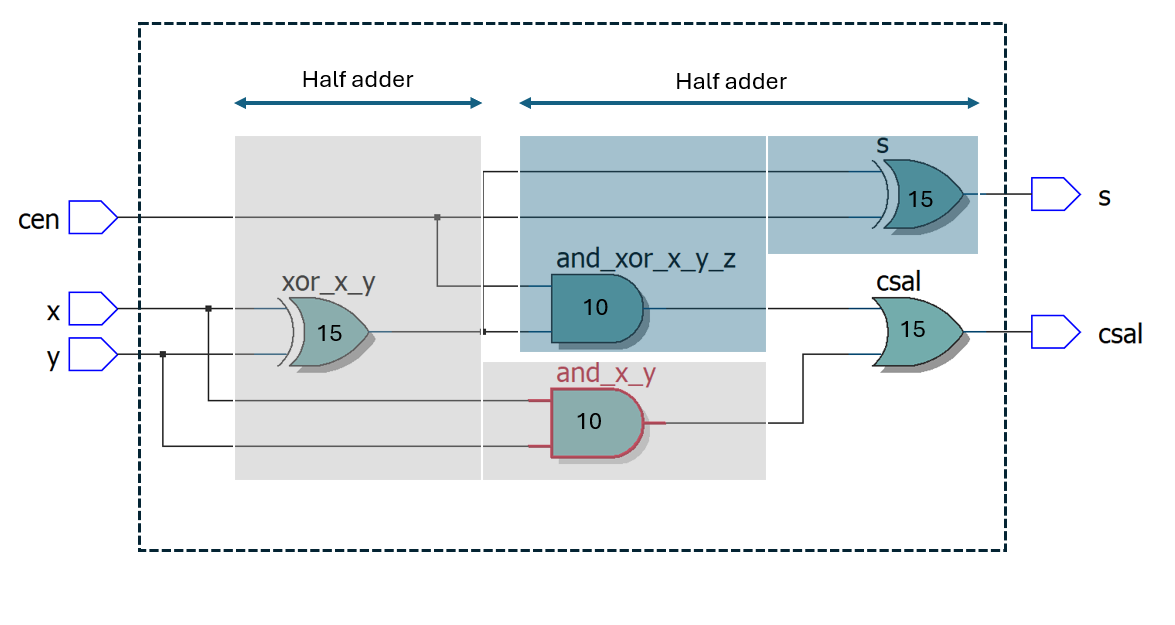


Cree un proyecto en Quartus. Para ello, cree el directorio ENTREGA en el directorio LAB1. Dentro del directorio ENTREGA cree los directorios descritos para almacenar los ficheros. Compruebe el funcionamiento lógico del esquema de circuito con Modelsim.

# **PREGUNTAS**

## **1. Elaboración RTL**

Entregue la elaboración RTL que efectúa Quartus del diseño efectuado



**Figura 1. RTL del sumador con retardos.**

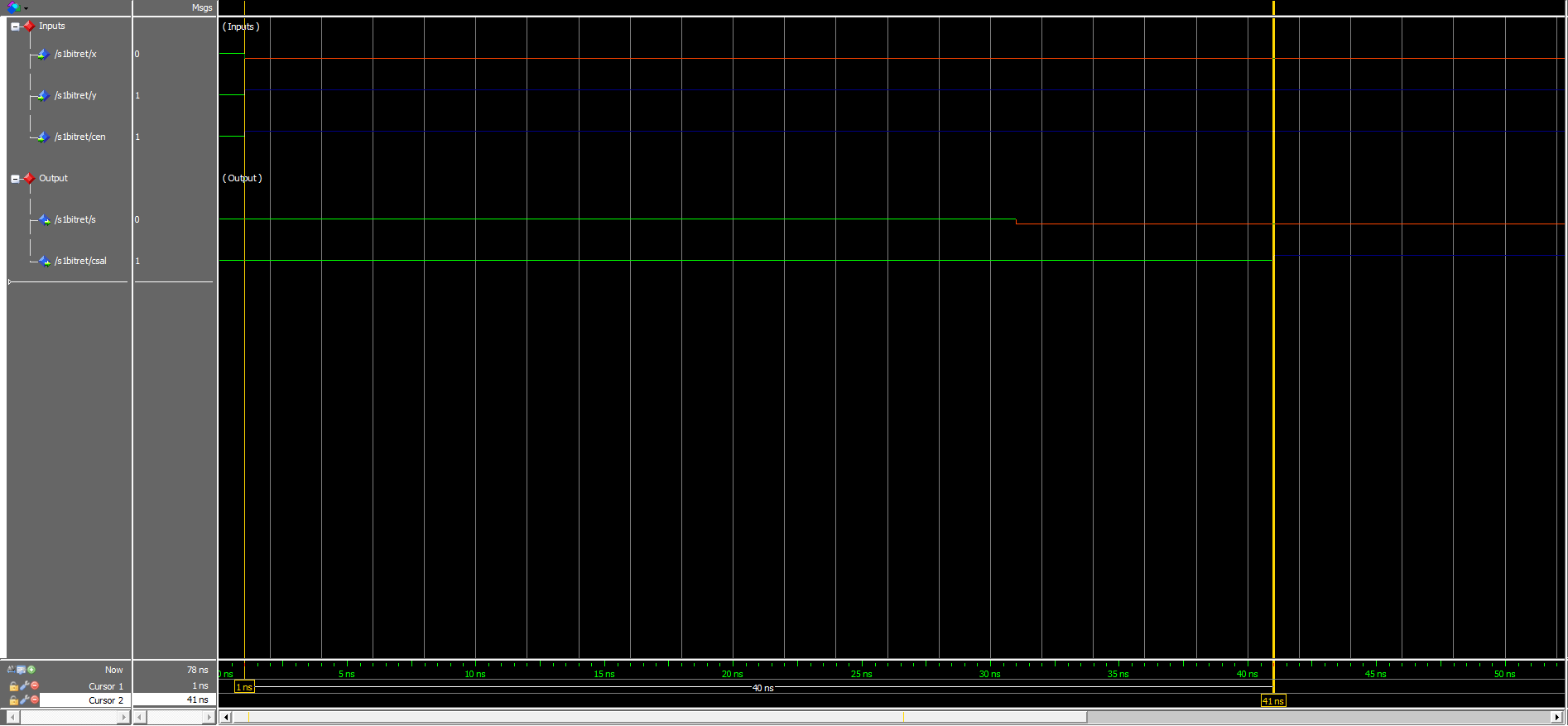
## **2. Creación fuente de onda**

Cree un frente de onda para medir el retardo de las 8 posibles combinaciones de los valores de las señales de entrada de un sumador de 1 bit. Utilice la orden force para estimular los puertos de entrada en el simulador Modelsim.

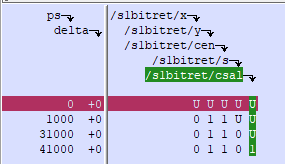
Ejemplo de frente de onda utilizado para X -> 0, Y -> 1 y Cen -> 1. Con delay de 1 ns y duración de 100 ns.

*force -freeze {sim:/s1bitret/x sim:/s1bitret/y sim:/s1bitret/cen} {0 1 1} 1ns, {0 0 0} 100ns*

Adjunte la parte de la ventana textual (list.lst) y la parte de la ventana temporal (wave.bmp) cuando se estimula el circuito con el vector de entrada (x = 0, y = 1, cen = 1). Muestre en la ventana textual el cálculo del retardo. Así mismo, identifique en la ventana temporal la evolución de las señales para este vector (utilice los cursores).



**Figura 2. Ventana Temporal de simulación**



**Figura 3. Ventana Textual**

## 

El retardo presentado en las 8 combinaciones se puede observar en la siguiente tabla (siendo los valores del vector de entrada (x,y,cen) en el orden establecido):

| Vector de entrada | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Retardo (ns) | 30 | 40 | 30 | 40 | 30 | 40 | 30 | 30 |
| Justificación:   1. En el diseño, el retardo de la señal **S** siempre será de **30 ns**. Esto se debe a que su cálculo depende de dos puertas XOR, cada una con un retardo de 15 ns, sumando un total de 30 ns para la salida de **S**. 2. Cuando **Cen** es 1, el retardo de la señal puede aumentar a **40 ns**. Esto ocurre porque la señal **and\_xor\_x\_y\_z** recibe un 1 y necesita esperar a la otra entrada, que es el resultado de la operación XOR entre **x** y **y**, la cual tiene un retardo adicional de 15 ns. Esto añade 10 ns de la operación AND más los 15 ns de la operación XOR, resultando en un total de 40 ns. Sin embargo, hay una excepción cuando **x** y **y** son ambos 1. En ese caso, la señal **csal** (carry out) se verá afectada por la operación OR, que no necesitará esperar el resultado de la otra entrada, ya que **and\_x\_y** proporcionará un 1. Con eso, la OR ya tiene suficiente información para determinar que el resultado será 1 sin esperar a la otra entrada. 3. Por otro lado, cuando **Cen** es 0, la señal **and\_xor\_x\_y\_z** será automáticamente 0, lo que significa que la salida **csal** no tendrá que esperar el resultado de la otra entrada. En este caso, el retardo total será de **25 ns**: 10 ns de las puertas AND y 15 ns de la puerta OR. Sin embargo, también existe una excepción en este caso. | | | | | | | | |

**Tabla 1. Tabla de retardos.**

## **3. Creación fuente de onda**

Indique, para cada valor del vector entrada E1, un vector de entrada previo E0 que

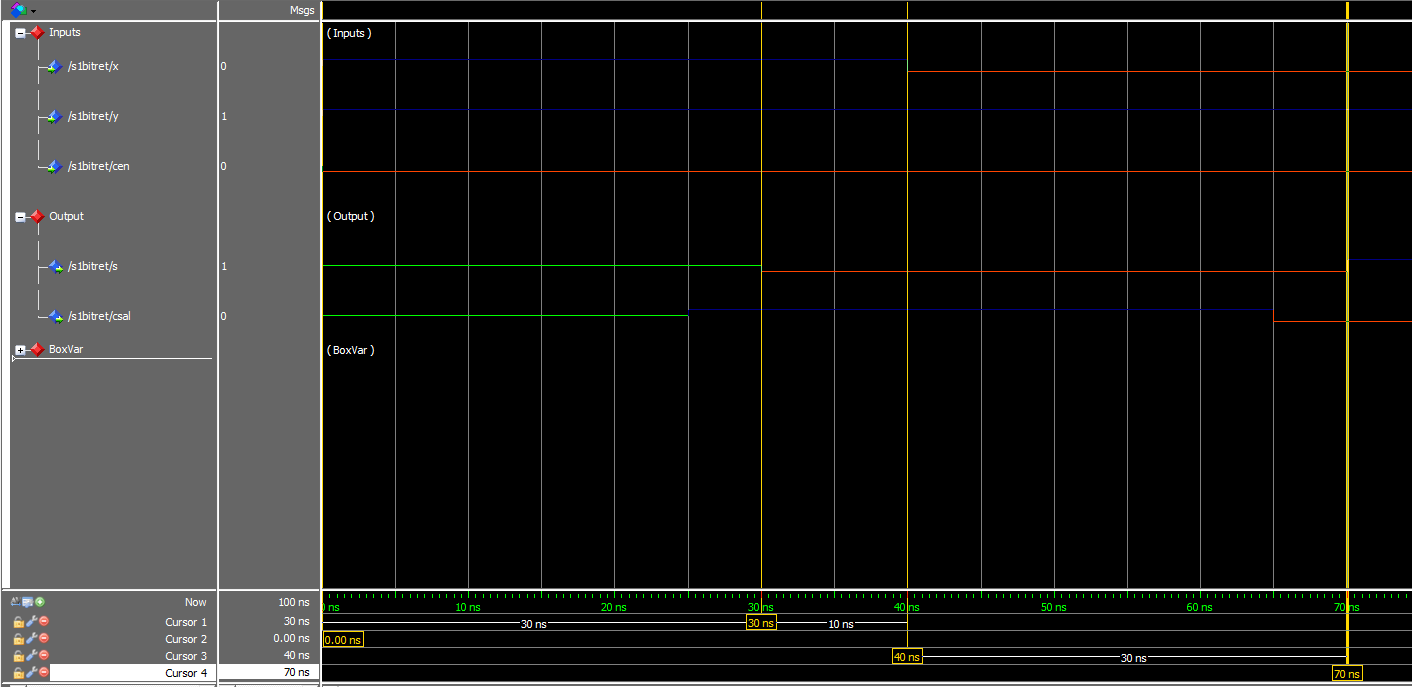
provoque el retardo (ret1) observado en la pregunta anterior. Para cada caso, minimice

el número de cambios en los bits de los vectores de entrada.

| E0 | | | E1 | | | ret observado |
| --- | --- | --- | --- | --- | --- | --- |
| x | y | cen | x | y | cen | ret 1 (ns) |
| 1 | 0 | 0 | 0 | 0 | 0 | 30 |
| 0 | 1 | 1 | 0 | 0 | 1 | 40 |
| 1 | 1 | 0 | 0 | 1 | 0 | 30 |
| 0 | 0 | 1 | 0 | 1 | 1 | 40 |
| 1 | 1 | 0 | 1 | 0 | 0 | 30 |
| 0 | 0 | 1 | 1 | 0 | 1 | 40 |
| 1 | 1 | 1 | 1 | 1 | 0 | 30 |
| 0 | 1 | 0 | 1 | 1 | 1 | 30 |

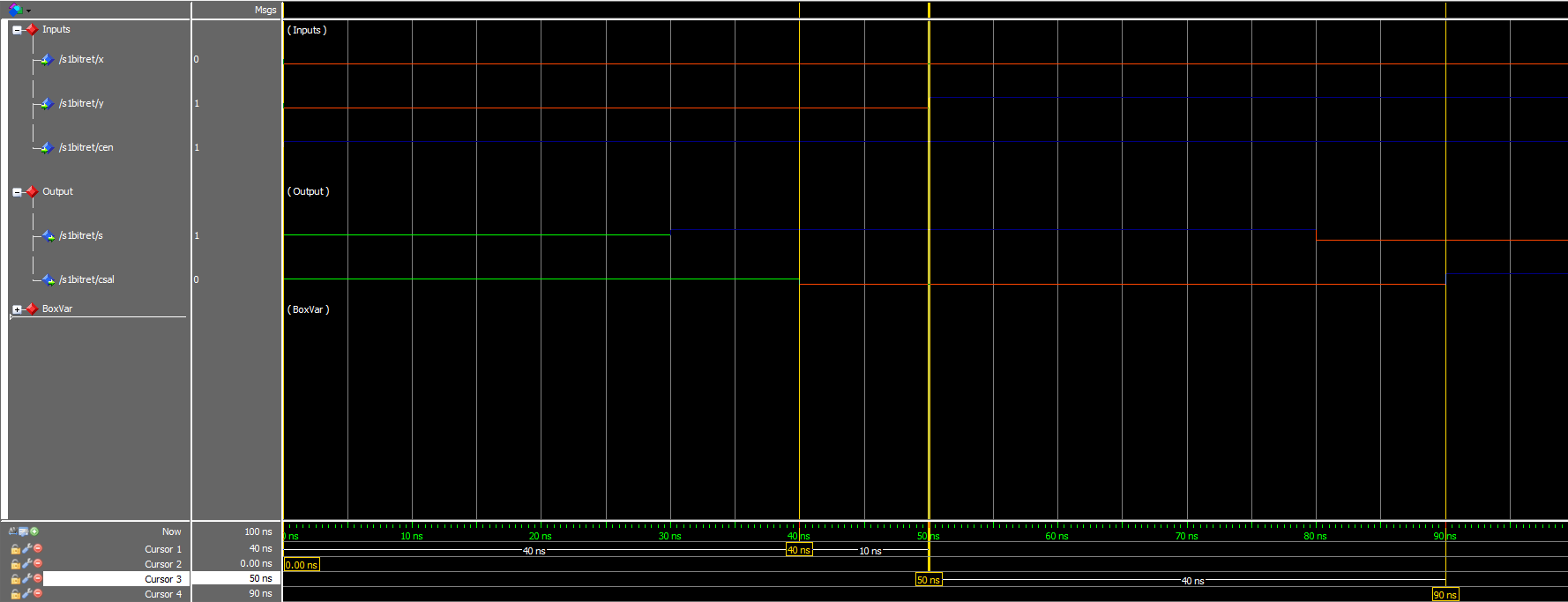
**Tabla 2. Fuentes de onda**

### **Caso A**



**Figura 4. Frente de onda caso A**

### **Caso B**



**Figura 5. Frente de onda caso B**

Se tiene que variar como mínimo un bit de entrada para cambiar de estado las salidas. El periodo mínimo de cambio de estados ha de ser mayor que el retardo máximo, en otro caso el sumador no funcionará correctamente.