Taller de Lógica Digital - Parte 2

Organización del Computador 1

Primer Cuatrimestre 2023

Ejercicios

1. Componentes de 3 estados

a) Completar la siguiente tabla:

A	A_{en}	В	B_{en}	\mathbf{C}	C_{en}	Estimado	Obtenido
0	0	0	0	0	0	Z	Z
0	1	1	1	0	0	CONFLICTO	CONFLICTO
1	0	1	0	1	0	Z	Z
1	1	0	0	0	1	CONFLICTO	CONFLICTO
0	1	0	1	0	1	0	0
0	1	1	1	1	1	CONFLICTO	CONFLICTO
1	0	1	1	1	0	1	1

b) Completar la siguiente tabla:

Color	Interpretación			
Gris	Hay algo que falta conectar			
Verde claro	Cuando hay algun enable activo y el dato es 1			
Verder oscuro	Cuando hay algun enable activo y el dato es 0			
Azul	Cuando todos los enables son 0 (Res = Z)			
Rojo	Cuando los enables activos se contradicen			
	1			

c) Enunciar la regla:

Si hay mas de un enable activo, que tengan el mismo dato de entrada.

d) Explicar cuáles son y por qué:

Los casos en los que atras del buffer algo que tira un Z y este entra como dato de entrada al buffer (Z como dato de entrada). Es algo que se puede hacer en el simulador, pero no vemos ningun caso por el estilo. Esto porque si Z está como entrada no afecta al comportamiento del programa.

2. Transferencia entre registros

a) Detallar entradas y salidas:

.clk: El dato por almacenar

.Force_input (control): Es el clk del registro_salida_restrictiva y su funcion es junto al w permitir el almacenamiento o no. Además cuando este se convierte en 1 hay un flanco de subida y ahí se almacena el dato.

.en_Force_Input (control): Es el enable del Force_input y nos permite cortar la señal de este (dando lugar a la señal de alta impedancia) o rehabiltarla (gerenado un flanco de subida en caso de que Force_input esté en 1).

.w (control): cada w controla junto al Force_input en su respectivo registro_salida_restringida el almacenamiento del dato (w tambien puede generar el flanco de subida). .en_out (control): cada en_out controla el paso por el Reg_output del dato almacenado.

b) Secuencia de señales:

Para activar el R1 pasamos el clk a 1 (dato a almacenar), el w a 1 (entrada de control de R1 para que se almacene el dato), dejamos pasar la corriente de Force_input con en_Force_input en 1 y habilitamos Force_input para generar un flanco de subida y que se almacene el dato.

Para desactivar, tenemos que pasar el clk a 0 y reiniciar el Force_input para generer otro flanco de subida (pasando de 1 a 0 y de 0 a 1 el mismo Force_Input o el en_Force_input) para almacenar el nuevo dato.

\	C		1	~ 1	
c	Secuen	Cla	de	sena	69

Teniendo un valor cualquiera en R0, ponemos ese valor en clk, prendemos Force_input y en_Force_input para que el clk de todos los registro_salida_restringida este en 1 siempre. Despues, prendemos el w de R1 para generar un flanco de subida y que se almacene el dato del clk. Ahora para pasar el dato de R2 a R0 tenemos que poner en el clk el valor de R2 y prender el w de R0 para generar el flanco de subida y que su dato almacenado pase a ser el del clk. Por ultimo, para pasar el R1 al R2 ponemos en el clk el valor de R1 y prendemos el w de R2 para generar el flanco de subida y que se almacene el dato del clk.

3. Máquina de 4 registros con suma y resta.

- a) Detallar entradas y salidas:
- b) Detallar el contenido de cada display:
- c) Secuencia de señales:
- d) Completar la siguiente tabla:

Valor inicial	Resultado operación 1	Flags	Resultado operación 2	Flags
(4, 0)				
(7, -1)				
(-8, -2)				
(8, -9)				

Los resultados interpretados en sin signo y en complemento a 2.

e) Explicar

Corrección

Integrantes:

Nombre y Apellido: LU: Nombre y Apellido: LU:

Para uso de los docentes:

1	2	3