

# 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 <sub>en</sub>	B	B <sub>en</sub>	C	C <sub>en</sub>	Estimado	Obtenido
0	0	0	0	0	0	Hi-Z	Hi-Z
0	1	1	1	0	0	error	error
1	0	1	0	1	0	Hi-Z	Hi-Z
1	1	0	0	0	1	error	error
0	1	0	1	0	1	0	0
0	1	1	1	1	1	error	error
1	0	1	1	1	0	1	1

b) Completar la siguiente tabla:

Color	Interpretación
Gris	Cable no conectado a nada
Verde claro	1
Verde oscuro	0
Azul	Hi-Z
Rojo	error

c) Enunciar la regla:

No se deben enviar un 0 y un 1 en dos entradas diferentes (A, B, C) cuando sus entradas de control son 1.

d) Explicar cuáles son y por qué:

Los combinaciones basura son aquellas en las que las entradas de control son 1 y los valores de entrada son iguales ya que se están enviando a la vez varias señales.

### 2. Transferencia entre registros

a) Detallar entradas y salidas:

entradas: clk representa el clock, Force\_Input es una entrada de data, en\_Force\_Input es la entrada de control que permite que pase el Force\_input, w es la entrada de write y en\_out la entrada que permite la salida del registro  
salidas: R0, R1 y R2 son las salidas de debug de los registros y la salida de la derecha es la salida de los registros

clk, en\_Force\_Input, w y en\_out son las entradas de control del circuito

b) Secuencia de señales:

Suponiendo que el clk se mueve constantemente:

w del registro R1 en 1 -> Force\_Input en 1 -> en\_Force\_Input en 1 -> esperar flanco ascendente de clk  
-> w del registro R1 en 0 -> en\_Force\_Input en 0 -> Force\_Input en 0

c) Secuencia de señales:

Suponiendo que el clk se mueve constantemente:

activo w de R0 -> pongo Force\_Input en el valor deseado -> activo en\_Force\_Input -> espero clk -> desactivo w de R0  
-> desactivo en\_Force\_Input -> activo en\_out de R0 -> activo w de R1 -> espero clk -> desactivo w de R1 -> desactivo en\_out de R0  
-> activo en\_out de R2 -> activo w de R0 -> espero clk -> desactivo w de R0 -> desactivo en\_out de R2 -> activo en\_out de R1  
-> activo w de R2 -> espero clk -> desactivo w de R2 -> desactivo en\_out de R1

### 3. Máquina de 4 registros con suma y resta.

a) Detallar entradas y salidas:

Con respecto a las entradas, ALU\_A\_Write es la entrada de control que permite la entrada de data al registro A del ALU, ALU\_B\_Write es la entrada de control que permite la entrada de data al registro B del ALU, OP es una entrada que permite controlar lo que hace la ALU con la data (suma, resta, or, and), ALU\_enableOut es la entrada que permite que salga la data del ALU, los Reg\_Write permiten escribir en los diferentes registros, los Reg\_enableOut permiten la salida de cada registro, Force\_Input es la entrada para ingresar un dato, en\_Force\_Input es la entrada para dejar pasar el dato de entrada a los registros, y clk es la entrada de clock.  
Las entradas de control son todas menos el Force\_Input.  
Con respecto a las salidas, N es la salida que dice cuando es un numero negativo, Z es la salida que dice cuando es 0, V es la salida que dice cuando hay overflow, y C es la salida que dice cuando hay carry o borrow.

b) Detallar el contenido de cada display:

El display de abajo del todo es el que indica el valor del Force\_Input o el output del ALU, los 4 displays de los Reg\_Debug muestran la salida debug de cada registro, el display del A\_Debug muestra el valor de A en el ALU, el display del B\_Debug muestra el valor de B en el ALU, y el display de S\_Debug muestra en todo momento el resultado del ALU.

c) Secuencia de señales:

Suponiendo que el clk se mueve constantemente:

ingreso el 4 (0100) en el Force\_Input -> activo el en\_Force\_Input -> activo Reg2\_Write -> espero clk -> desactivo Reg2\_Write  
-> ingreso el -3 (1101) en el Force\_Input -> activo Reg3\_Write -> espero clk -> desactivo Reg3\_Write -> desactivo en\_Force\_Input

d) Completar la siguiente tabla:

	Valor inicial	Resultado operación 1	Flags	Resultado operación 2	Flags
d) secuencia 1er caso "(4,0), OR, SUB": Force_Input = 0100 -> en_Force_Input = 1 -> Reg0_Write = 1 -> clk = 1 -> Reg0_Write = 0 -> en_Force_Input = 0 -> Reg0_enableOut = 1 -> ALU_A_Write = 1 -> clk = 1 -> ALU_A_Write = 0 -> Reg0_enableOut = 0 -> Force_Input = 0000 -> en_Force_Input = 1 -> Reg1_Write = 1 -> clk = 1 -> Reg1_Write = 0 -> en_Force_Input = 0 -> Reg1_enableOut = 1 -> ALU_B_Write = 1 -> clk = 1 -> ALU_B_Write = 0 -> Reg1_enableOut = 0 -> OP = 11 -> ALU_enableOut = 1 -> Reg2_Write = 1 -> clk = 1 -> Reg2_Write = 0 -> OP = 01 -> Reg3_Write = 1 -> clk = 1 -> Reg3_Write = 0	(4, 0)	0100 / 4 / 4	-	0100 / 4 / 4	-
	(7, -1)	1000 / 8 / -8	N V C	0111 / 7 / 7	-
	(-8, -2)	0110 / 6 / 6	V C	1010 / 10 / -6	N C
	(8, -9)	no son representables, el circuito toma en complemento a 2	-	-	-

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